



低レイテンシーストリーミングユーザーガイド

Amazon Interactive Video Service



Amazon Interactive Video Service: 低レイテンシーストリーミングユーザーガイド

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon の商標およびトレードドレスは、Amazon のものではない製品またはサービスと関連付けてはならず、また、お客様に混乱を招くような形や Amazon の信用を傷つけたり失わせたりする形で使用することはできません。Amazon が所有していない他のすべての商標は、それぞれの所有者の所有物であり、Amazon と提携、接続、または後援されている場合とされていない場合があります。

Table of Contents

IVS 低レイテンシーストリーミングとは	1
レイテンシー	1
グローバルソリューション、リージョナルコントロール	2
ストリーミングと視聴はグローバル	2
コントロールはリージョナル	2
チャンネルのリージョン	3
IVS の開始方法	4
ステップ 1: AWS アカウントの作成	4
ステップ 2: ルートユーザーと管理ユーザーのセットアップ	5
AWS アカウントのルートユーザーを保護する	5
管理ユーザーを作成する	5
ステップ 3: IAM アクセス許可の設定	6
IVS のアクセス許可には既存のポリシーを使用してください。	6
オプション: Amazon IVS アクセス許可の新しいポリシーを作成する	7
新しいユーザーを作成し、アクセス許可を付与する	9
既存のユーザーへのアクセス許可を追加する	10
ステップ 4: 任意の録画によるチャンネルの作成	10
Amazon S3 への自動録画	11
コンソールでの手順	12
CLI の手順	19
ステップ 5: ストリーミングソフトウェアの設定	23
Amazon IVS Broadcast SDK によるストリーミング	24
Amazon IVS コンソールを使用したストリーミング	24
OBS Studio によるストリーミング	25
FFmpeg による録画動画のストリーミング	27
ステップ 6: ライブストリームの表示	28
ステップ 7: Service-Quota 制限の確認 (オプション)	28
ステップ 8: 望ましくないコンテンツや視聴者を防ぐ (推奨)	29
コンソールの手順 (プレイバック制限ポリシー)	29
CLI の手順 (Playback 制限ポリシー)	30
録画を無効にする方法	31
IVS ストリームで複数ホストを有効にする	32
開始方法	32
コンソールでの手順	32

CLI の手順	34
ステージのブロードキャスト: クライアントサイトとサーバーサイドのコンポジションの対 比	35
デモ	36
1. ステージの作成	37
2. 参加者トークンの配布	38
3. ステージに参加する	38
4. ステージをブロードキャストする	41
モニタリング	45
前提条件	45
ストリームセッションデータへのアクセス	46
コンソールでの手順	46
AWS SDK での手順	47
CLI の手順	48
ヘルスによるストリームのフィルタリング	49
コンソールでの手順	50
CLI の手順	50
ConcurrentStreams の CloudWatch ヘルスディメンション	50
CloudWatch メトリクスへのアクセス	50
CloudWatch コンソールでの手順	51
CLI の手順	52
CloudWatch メトリクス: IVS 低レイテンシーストリーミング	52
IVS Broadcast SDK	58
プラットフォームの要件	58
ネイティブプラットフォーム	58
デスクトップブラウザ	59
モバイルブラウザ	60
ウェブビュー	60
必要なデバイスのアクセス	60
サポート	60
バージョンニング	61
Web ガイド	62
.....	62
開始方法	62
既知の問題と回避策	68
Android ガイド	71

開始方法	72
高度なユースケース	75
既知の問題と回避策	81
iOS ガイド	82
開始方法	83
高度なユースケース	89
iOS がカメラの解像度とフレームレートを選択する方法	98
既知の問題と回避策	98
ミキサーガイド	99
用語	100
キャンバスプロパティ	102
スロットのプロパティ	103
ミキシング用のブロードキャストセッションの設定	107
スロットを追加する	109
スロットの削除	110
トランジションのあるアニメーション	110
ブロードキャストのミラーリング	112
カスタム画像ソース	114
Android	114
iOS	115
IVS プレーヤー SDK	117
ブラウザとプラットフォームの要件	118
デスクトップブラウザ	119
モバイルブラウザ	120
ネイティブプラットフォーム	120
サードパーティ製プレーヤーのレイテンシーの削減	121
iOS Safari	121
オーディオのみの再生	121
サポート	122
バージョニング	122
Web ガイド	123
開始方法	123
フレームワークの統合	127
コンテンツセキュリティポリシーの操作	127
既知の問題と回避策	128
Android ガイド	129

開始方法	130
SDK サイズ	133
既知の問題と回避策	133
iOS ガイド	134
開始方法	134
SDK サイズ	141
既知の問題と回避策	141
Video.js 統合	141
開始方法	141
イベント	145
エラー	145
プラグイン	146
コンテンツセキュリティポリシー	147
関数	147
currentTime	149
dispose	150
duration	150
getIVSEvents	151
getIVSPlayer	151
load	152
play	152
playbackRate	153
seekable	153
JW Player 統合	154
開始方法	154
イベント	156
エラー	157
コンテンツセキュリティポリシー	158
制限事項	158
動画ストリーム内にメタデータを埋め込む	159
時間指定メタデータとは	159
IAM アクセス許可のセットアップ	159
時間指定メタデータの挿入	160
AWS CLI の使用	160
Amazon IVS API の使用	161
IVS Broadcast SDK の使用	161

時間指定メタデータの使用	161
サンプルデモ: クイズアプリ	162
時間指定メタデータの表示	163
追加情報	163
プライベートチャネルの設定	164
プライベートチャネルのワークフロー	165
再生キーの作成またはインポート	166
キーペアの作成	166
既存のパブリックキーをインポートするには	167
API リクエスト	168
チャンネルでの再生承認を有効にする	168
コンソールの手順	168
CLI の手順	169
API リクエスト (作成と更新)	170
再生トークンの生成と署名	170
トークンスキーマ	170
手順	173
再生キーのリスト化	173
コンソールの手順	173
CLI の手順	173
API リクエスト	174
再生キーの削除	174
コンソールの手順	174
CLI の手順	175
API リクエスト	175
再生キー情報の取得	175
コンソールの手順	176
CLI の手順	176
API リクエスト	176
ビューアーセッションを取り消す	176
CLI の手順	177
API リクエスト	177
Amazon S3 への自動録画	178
S3 プレフィックス	178
録画の内容	179
バイト範囲プレイリスト	180

サムネイル	180
断片化されたストリームの結合	181
対象	181
既知の問題	182
JSON メタデータファイル	182
例: recording_started.json	187
例: recording_ended.json	188
例: recording_failed.json	190
録画のレンディションの検出	191
プライベートバケットからの録画コンテンツの再生	193
Amazon CloudFront デイストリビューション	193
Amazon からの再生 CloudFront	194
IVS EventBridge での Amazon の使用	196
Amazon IVS の Amazon EventBridge ルールの作成	200
例: ストリーム状態の変化	200
例: ストリームの正常性の変化	202
例: 制限違反	203
例: 録画状態の変化	206
AWS CloudTrail を使用した API コールのログ記録	209
CloudTrail 内の Amazon IVS に関する情報	209
Amazon IVS ログファイルエントリの概要	210
セキュリティ	213
データ保護	214
ID とアクセス管理	215
対象者	215
Amazon IVS で IAM を使用する方法	216
ID	216
ポリシー	216
Amazon IVS タグに基づく承認	217
ロール	218
特権アクセスと非特権アクセス	218
ポリシーのベストプラクティス	218
アイデンティティベースのポリシーの例	219
トラブルシューティング	223
Amazon IVS 用マネージドポリシー	225
IVSReadOnlyAccess	226

IVSFullAccess	226
ポリシー更新	226
サービスリンクロールの使用	227
Amazon IVS のサービスにリンクされたロールのアクセス権限	228
Amazon IVS のサービスにリンクされたロールの作成	228
Amazon IVS のサービスにリンクされたロールの編集	229
Amazon IVS のサービスにリンクされたロールの削除	229
Amazon IVS サービスにリンクされたロールがサポートされるリージョン	230
ログ記録とモニタリング	230
インシデントへの対応	230
レジリエンス	230
Amazon IVS 動画データプレーン	231
インフラストラクチャセキュリティ	231
API 呼び出し	231
ストリーミングおよび再生	232
Service Quotas	233
サービスクォータの引き上げ	233
API コールレートクォータ	233
その他のクォータ	235
Service Quotas と CloudWatch 使用状況メトリクスの統合	239
使用状況メトリクスの CloudWatch アラームの作成	241
ストリーミング設定	242
前提条件	242
レイテンシーの低減	242
サードパーティによるストリーミング/転送サービスの回避	243
エンコーダー設定	244
ストリームの取り込み: コーデック、RTMPS、および 443 番ポート	244
解像度/ビットレート/FPS	244
チャンネルタイプ	245
ビデオ設定	251
オーディオ設定	251
VBR ではなく CBR を使用する	252
プログレッシブシグナルを使用する	252
ネットワークの要件	252
クローズドキャプション	253
FFmpeg を使用したストリーミング	254

Amazon IVS Broadcast SDK を使用してストリーミングする	255
ストリームのテスト	255
トラブルシューティングに関するよくある質問	256
ブロードキャストとエンコーディング	256
ストリームスタベーションとは何ですか?	257
ストリームが突然停止するのはなぜですか?	257
ストリーミング中にネットワークを切り替えるとどうなりますか?	258
IVS でマルチリージョンの冗長性を実現するにはどうしたらいいですか?	259
IVS Web Broadcast SDK セッションをトラブルシューティングするにはどうすればよいですか?	260
Google Chrome の WebRTC-internals メトリクスを使用して IVS Web Broadcast SDK セッションを評価するにはどうすればよいですか?	261
モニタリングとイベント	262
ストリームスタベーションのイベントをモニタリングするにはどうすればよいですか?	263
Amazon を使用して IVS サービスクォータ CloudWatch をモニタリングするにはどうすればよいですか?	264
IVS Stream Health を使用してストリームの不安定性を診断するにはどうすればよいですか?	264
ストリーム再生	272
IVS プレイヤーの動作をデバッグするにはどうすればよいですか?	272
すべての視聴者の再生がフリーズ/停止したのはなぜですか?	272
IVS プレイヤーがバッファリングする原因は何ですか?	273
Amazon S3 への自動録画	273
一部の録画コンテンツが欠落しているのはなぜですか?	274
KMS-S3 暗号化は S3 への自動録画で使用できますか?	274
その他のトピック	274
「pending verification」(検証待ち) エラーは何を意味していますか?	275
IVS の使用コストを見積もることはできますか?	275
望ましくないコンテンツと視聴者	276
望ましくないコンテンツの検出	276
異常検出	276
カスタムコンテンツモデレーション	277
望ましくないコンテンツや視聴者の防止	278
ストリームを停止して、ストリームキーをリセットする	278
プライベートチャンネルを使用する	278
再生制限ポリシーを使用する	279

コスト	280
ライブ動画	280
Amazon S3 への自動録画	281
録画した動画の保存	281
録画した動画の提供	284
リソースおよびサポート	285
リソース	285
デモ	285
パートナーソリューション	285
分析	286
インタラクティビティ	286
フェイスフィルターと背景フィルター	286
サポート	287
用語集	288
ドキュメント履歴	309
低レイテンシーストリーミングユーザーガイドの変更	309
IVS 低レイテンシーストリーミング API リファレンスの変更	410
ステージ API リファレンスの変更	418
IVS チャット API ドキュメントの変更	419
リリースノート	421
2024 年 2 月 1 日	421
Amazon IVS Broadcast SDK: Android 1.14.1、iOS 1.14.1、Web 1.8.0 (低レイテンシーストリーミング)	421
2024 年 1 月 31 日	422
トークンレス再生の制限	422
2024 年 1 月 25 日	422
オーディオのみの再生	422
2024 年 1 月 18 日	423
Amazon IVS Player SDK 1.24.0	423
2024 年 1 月 3 日	424
Amazon IVS Broadcast SDK: Android 1.13.4、iOS 1.13.4、Web 1.7.0 (低レイテンシーストリーミング)	424
2023 年 12 月 4 日	426
Amazon IVS Broadcast SDK: Android 1.13.2、iOS 1.13.2 (低レイテンシーストリーミング)	426
2023 年 11 月 21 日	427

Amazon IVS Broadcast SDK: Android 1.13.1 (Low-Latency Streaming)	427
2023 年 11 月 17 日	428
Amazon IVS Broadcast SDK: Android 1.13.0、iOS 1.13.0 (Low-Latency Streaming)	428
2023 年 11 月 14 日	430
Amazon IVS Player SDK 1.23.0	430
2023 年 10 月 16 日	431
Amazon IVS Broadcast SDK: Web 1.6.0 (Low-Latency Streaming)	431
2023 年 10 月 12 日	432
Amazon IVS Broadcast SDK: Android 1.12.1 (Low-Latency Streaming)	432
2023 年 10 月 3 日	432
Amazon IVS Player SDK 1.22.0	432
2023 年 10 月 2 日	434
コンソール内ストリーミング	434
2023 年 9 月 14 日	434
Amazon IVS Broadcast SDK: Web 1.5.2 (Low-Latency Streaming)	434
2023 年 8 月 23 日	435
Amazon IVS Broadcast SDK: Web 1.5.1、Android 1.12.0、iOS 1.12.0 (Low-Latency Streaming)	435
2023 年 8 月 23 日	436
Amazon IVS Broadcast SDK: Android 1.7.6 (Low-Latency Streaming)	436
2023 年 8 月 22 日	437
Amazon IVS Player SDK 1.21.0	437
2023 年 8 月 7 日	439
Amazon IVS Broadcast SDK: Web 1.5.0、Android 1.11.0、iOS 1.11.0	439
2023 年 7 月 17 日	440
R2S3 レンダيشョンのフィルタリングおよびサムネイルの強化	440
2023 年 7 月 14 日	441
Amazon IVS Player SDK 1.20.0	441
2023 年 7 月 13 日	443
Amazon IVS Broadcast SDK: Web 1.4.0、Android 1.10.0、および iOS 1.10.0	443
2023 年 6 月 28 日	446
プライベートチャンネルのビューアーセッションの取り消し	446
2023 年 6 月 27 日	447
Amazon IVS Broadcast SDK: iOS 1.9.1	447
2023 年 6 月 27 日	448
Amazon IVS Broadcast SDK 1.7.5	448

2023 年 6 月 16 日	449
Amazon IVS Broadcast SDK: Web 1.3.3	449
2023 年 6 月 2 日	449
Advanced チャンネルタイプ	449
2023 年 6 月 1 日	450
Amazon IVS Broadcast SDK: Android 1.9.0 および iOS 1.9.0	450
2023 年 5 月 23 日	453
Amazon IVS Player SDK 1.19.0	453
2023 年 5 月 16 日	454
Amazon IVS Broadcast SDK: iOS 1.8.1	454
2023 年 5 月 16 日	455
Amazon IVS Broadcast SDK 1.7.4	455
2023 年 5 月 11 日	456
複数ホストのヘルス	456
2023 年 5 月 1 日	456
Amazon IVS Web Broadcast SDK 1.3.2	456
2023 年 4 月 27 日	457
ステージ参加者の増加	457
2023 年 4 月 4 日	457
Amazon IVS Player SDK 1.18.0	457
2023 年 3 月 30 日	459
RTMP サポート	459
2023 年 3 月 29 日	459
プライベートチャンネル用の使い捨てトークン	459
2023 年 3 月 28 日	460
Amazon IVS Web Broadcast SDK 1.3.1	460
2023 年 3 月 23 日	460
ストリーム上の複数ホストのサポート (ステージリソース)	460
2023 年 3 月 23 日	460
Amazon IVS Broadcast SDK: Android 1.8.0、iOS 1.8.0、Web 1.3.0	460
2023 年 3 月 2 日	463
Amazon IVS Broadcast SDK: Android 1.7.3	463
2023 年 2 月 28 日	464
Amazon IVS Player SDK 1.17.0	464
2023 年 2 月 16 日	465
S3 への自動録画用のバイト範囲タグとマニフェストファイル	465

2023 年 1 月 31 日	465
Amazon IVS Chat Client Messaging SDK: Android 1.1.0	465
2023 年 1 月 17 日	466
Amazon IVS Player SDK 1.16.0	466
2022 年 12 月 9 日	468
S3 マニフェストファイルへの自動記録に追加されたタイムスタンプ	468
2022 年 12 月 6 日	468
Amazon IVS Broadcast SDK: Android 1.7.2	468
2022 年 11 月 17 日	469
チャットのログ記録	469
2022 年 11 月 9 日	470
Amazon IVS Chat Client Messaging SDK: JavaScript 1.0.2	470
2022 年 11 月 1 日	470
Amazon IVS Player SDK 1.14.0	470
2022 年 10 月 18 日	472
Amazon IVS Chat Client Messaging SDK: JavaScript 1.0.1	472
2022 年 10 月 6 日	472
Amazon IVS Broadcast SDK 1.7.1	472
2022 年 9 月 22 日	473
Amazon IVS Broadcast SDK 1.7.0	473
2022 年 9 月 20 日	475
Amazon IVS Player SDK 1.13.0	475
2022 年 9 月 15 日	477
縦型動画の向上 (最終リリース)	477
2022 年 9 月 12 日	477
Amazon IVS Broadcast SDK 1.5.2: iOS	477
2022 年 9 月 8 日	478
Amazon IVS Chat Client Messaging SDK: Android 1.0.0 および iOS 1.0.0	478
2022 年 9 月 2 日	479
Amazon IVS Web Broadcast SDK 1.2.0	479
2022 年 8 月 30 日	479
断片化されたストリームの結合	479
2022 年 8 月 9 日	480
Amazon IVS Web Player SDK 1.12.0	480
2022 年 7 月 28 日	481
Amazon IVS iOS Broadcast SDK 1.5.1	481

2022 年 7 月 21 日	482
Amazon IVS Web Broadcast SDK	482
2022 年 7 月 14 日	482
Amazon IVS iOS Player SDK 1.8.3	482
2022 年 6 月 28 日	483
Amazon IVS Player Web SDK 1.11.0	483
2022 年 6 月 22 日	484
Amazon IVS Broadcast SDK 1.5.0	484
2022 年 6 月 9 日	486
垂直方向の動画の改善	486
2022 年 5 月 24 日	488
Amazon IVS Web および Android Player SDK 1.10.0	488
2022 年 4 月 28 日	490
ストリームヘルスの更新	490
2022 年 4 月 26 日	490
Amazon IVS Chat	490
2022 年 4 月 22 日	491
Amazon IVS iOS Player SDK 1.8.2	491
2022 年 4 月 19 日	492
Amazon IVS Broadcast SDK 1.4.0	492
2022 年 3 月 31 日	494
Amazon IVS iOS Player SDK 1.8.1	494
2022 年 3 月 3 日	495
Amazon IVS Broadcast SDK 1.3.0	495
2022 年 3 月 1 日	497
Amazon IVS Player SDK 1.8.0	497
2022 年 2 月 3 日	499
Amazon IVS Broadcast SDK: Android 1.2.1	499
2022 年 1 月 20 日	500
Amazon IVS Player SDK 1.7.0	500
2022 年 1 月 18 日	504
R2S3 サムネイル設定	504
2021 年 12 月 9 日	504
Amazon IVS Broadcast SDK 1.2.0	504
2021 年 11 月 23 日	507
Amazon IVS Player SDK 1.6	507

2021 年 11 月 18 日	510
Stream Health	510
2021 年 10 月 20 日	511
Amazon IVS Broadcast SDK: Android 1.1.0 および iOS	511
2021 年 9 月 29 日	515
Amazon IVS Player SDK: Android 1.5.1	515
2021 年 9 月 28 日	515
Amazon IVS Player SDK 1.5.0	515
2021 年 9 月 8 日	519
Amazon IVS Player SDK 1.4.1	519
2021 年 8 月 13 日	521
ListTagsForResource API エンドポイント	521
2021 年 8 月 10 日	521
Amazon IVS Player SDK 1.4.0	521
2021 年 7 月 27 日	526
Amazon IVS Broadcast SDK: Android 1.0.0 および iOS 1.0.0	526
2021 年 6 月 1 日	527
Amazon IVS Player SDK: Android 1.3.3 と iOS 1.3.3	527
2021 年 5 月 19 日	527
Amazon IVS Player SDK: Android 1.3.2	527
2021 年 5 月 5 日	528
Amazon IVS Player SDK 1.3	528
2021 年 4 月 26 日	533
Service Quotas と CloudWatch 使用状況メトリクスの統合	533
2021 年 4 月 13 日	533
新しい CloudWatch メトリクス	533
2021 年 4 月 7 日	533
S3 への自動録画 (R2S3)	533
2021 年 1 月 28 日	534
Amazon IVS Player SDK: JW Player 統合 1.2.0	534
2020 年 12 月 16 日	534
Amazon IVS Player: SDK for Android 1.2.1	534
2020 年 11 月 23 日	534
Amazon IVS Player SDK 1.2.0	534
2020 年 11 月 12 日	536
新しいイベントフィールド、stream_id	536

2020 年 11 月 9 日	537
コンソールへのメタデータ表示の追加	537
2020 年 10 月 30 日	537
CloudFormation サポート	537
2020 年 10 月 27 日	537
チャンネル、CCV、CCB 制限の引き上げ	537
2020 年 10 月 9 日	538
新しい Service Quotas と EventBridge イベント	538
Amazon IVS Player: SDK for Web 1.1.2	538
2020 年 10 月 7 日	538
Amazon IVS Player SDK 1.1.0	538
2020 年 9 月 14 日	542
新しいイベントフィールド、channel_name	542
2020 年 8 月 19 日	542
再生承認 (プライベートチャンネル)	542
2020 年 8 月 11 日	543
Amazon IVS Player SDK: SDK for iOS 1.0.6	543
2020 年 8 月 5 日	543
Amazon IVS EventBridge での Amazon の使用	543
2020 年 7 月 15 日	543
プレイヤーバージョン 1.0	543
.....	dxlvi

Amazon IVS 低レイテンシーストリーミングとは

Amazon Interactive Video Service (IVS) は、マネージド型のライブビデオストリーミングサービスです。以下のことが行えます。

- チャンネルを作成して数分以内にストリーミングを開始する。
- ultra-low-latency ライブ動画と一緒に魅力的なインタラクティブエクスペリエンスを構築します。
- さまざまなデバイスやプラットフォーム向けに大規模に動画を配信する。
- ウェブサイトやアプリに簡単に統合できる。

Amazon IVS を使えば、独自のインタラクティブアプリケーションと視聴者体験の構築に集中できます。Amazon IVS を使うことで、インフラストラクチャの管理や、ビデオワークフローのコンポーネントの開発および設定が不要になり、安全性、安定性、費用対効果を確保できます。

Amazon IVS は、RTMPS および RTMP ストリーミングの両方をサポートしています。RTMPS は、TLS 上で実行される RTMP (リアルタイムメッセージングプロトコル) の安全なバージョンです。RTMP は、ネットワーク経由で動画を送信するための業界標準です。

製品に関するドキュメントに加えて、<https://ivs.rocks/> を参照してください。これは、公開済みコンテンツ (デモ、コードサンプル、ブログ投稿) を閲覧し、コストを見積もり、ライブデモを通じて Amazon IVS を体験するための専用サイトです。

レイテンシー

レイテンシーとは、カメラがライブストリームをキャプチャしてから視聴者の画面に表示されるまでの遅延のことです。Amazon IVS には、次のようにビデオを配信できる機能があります。

- 超低レイテンシー — Amazon IVS チャンネルは、5 秒未満のレイテンシーで動画を配信できます。
- リアルタイムレイテンシー — IVS ステージは、300 ミリ秒未満のレイテンシーで動画を配信できます。ステージの参加者全員が、この強化された「リアルタイムレイテンシー」を体験します (ステージが IVS チャンネルに配信される場合、チャンネル視聴者のレイテンシーは超低レイテンシーになります)。

従来のオーバーザトップ (OTT) ストリーミングの場合、レイテンシーは最大 30 秒になることもあります。

低レイテンシーは、視聴者体験を豊かにする、優れたインタラクティブユーザー体験を構築する上で欠かせない要素です。それにより、ストリーマー、ブランド、コミュニティは、ライブの視聴者と直接的かつパーソナルな方法でつながることが可能になります。

観測されるレイテンシーは、次の理由によりユーザーによって変化することがあります。

- ストリーマーと視聴者の地理的位置。
- ネットワークの種類と速度。
- ストリーミングチェーン内の個々のコンポーネント。
- ストリーミングプロトコルと出力形式。

詳細については、Amazon IVS Streaming Configuration の「[Reducing Latency](#)」を参照してください。

グローバルソリューション、リージョナルコントロール

ストリーミングと視聴はグローバル

Amazon IVS を使用して、世界中の視聴者にストリーミングできます。

- ストリーミングすると、Amazon IVS はユーザーの近くの場所で自動的に動画を取り込みます。
- 視聴者は、Amazon IVS コンテンツ配信ネットワークを介して、ライブストリームをグローバルに視聴できます。

つまり、「データプレーン」はグローバルです。データプレーンとは、ストリーミング/取り込み、視聴を指します。

コントロールはリージョナル

Amazon IVS データプレーンはグローバルですが、「コントロールプレーン」はリージョンに基づきます。コントロールプレーンとは、Amazon IVS コンソール、API、リソース (チャンネル、ストリームキー、再生キーペア、録画設定)を指します。

つまり、Amazon IVS は「リージョンによる AWS のサービス」です。各リージョンの Amazon IVS リソースは、他のリージョンの類似リソースから独立しています。たとえば、あるリージョンで作成したチャンネルは、他のリージョンで作成したチャンネルとは無関係です。

リソースを使用するときは (チャンネル作成時など)、リソースを作成するリージョンを指定する必要があります。その後、リソースを管理するときは、リソースを作成したリージョンと同じリージョンで管理する必要があります。

...を使用した場合	...によりリージョンを指定します。
Amazon IVS コンソール	ナビゲーションバー右上の [リージョンの選択] ドロップダウンを使用します。
Amazon IVS API	適切なサービスエンドポイントの使用 「 Amazon IVS 低遅延ストリーミング API リファレンス 」を参照してください。 (SDK を介して API にアクセスする場合は、SDK のregionパラメータをセットアップします。 AWS で構築するツール を参照してください)。
AWS CLI	次のいずれかを実行します: <ul style="list-style-type: none"> • CLI コマンドに <code>--region <aws-region></code> を追加します。 • ローカルの AWS 設定ファイルにリージョンを追加します。

チャンネルが作成されたリージョンにかかわらず、どこからでも Amazon IVS にストリーミングでき、視聴者はどこからでも視聴できます。

チャンネルのリージョン

チャンネルのリージョンは、チャンネルの作成時に割り当てられる ARN (Amazon リソースネーム) の一部です。チャンネル作成時:

- Amazon IVS コンソールは、ページの General configuration (一般設定) エリアの ARN を表示します。その後は、コンソールが右上にリージョンを常に表示します。
- Amazon IVS API は、チャンネルオブジェクトのarnフィールドに ARN を返します。

IVS 低レイテンシーストリーミングを開始する

このドキュメントでは、最初の Amazon Interactive Video Service (IVS) ライブストリームをセットアップするステップを説明します。

トピック

- [ステップ 1: AWS アカウントの作成](#)
- [ステップ 2: ルートユーザーと管理ユーザーのセットアップ](#)
- [ステップ 3: IAM アクセス許可の設定](#)
- [ステップ 4: 任意の録画によるチャンネルの作成](#)
- [ステップ 5: ストリーミングソフトウェアの設定](#)
- [ステップ 6: ライブストリームの表示](#)
- [ステップ 7: Service-Quota 制限の確認 \(オプション\)](#)
- [ステップ 8: 望ましくないコンテンツや視聴者を防ぐ \(推奨\)](#)
- [録画を無効にする方法](#)

ステップ 1: AWS アカウントの作成

Amazon IVS を使用するには、AWS アカウントが必要です。アカウントをまだお持ちでない場合は、サインアップ時に画面の指示に従って作成してください。AWS アカウントを作成するには:

1. <https://portal.aws.amazon.com/billing/signup> を開きます。
2. オンラインの手順に従います。

サインアップ手順の一環として、通話呼び出しまたはテキストメッセージを受け取り、確認コードを入力することが求められます。また、請求情報を提供する必要がありますが、ベーシックなサービス層は無料です。サインアップした AWS サービスの料金は、そのサービスを使用しない限り発生することはありません。

3. アカウントの作成後に、サインイン URL およびユーザー名が記載された電子メールが送信され、(AWS アカウント管理者から) パスワードが記載された別の電子メールが送信されます。初回サインイン時にパスワードを変更する必要があります。

既存の AWS アカウントを使用する場合は、Amazon IVS をサポートしている AWS リージョンを使用していることを確認してください。

1. [Amazon IVS コンソール](#)に移動します。(「グローバルソリューション、地域別コンテンツ」と表示されている) 通常の IVS コンソールページが表示されれば、問題ありません。「[ステップ 2: ルートユーザーと管理ユーザーのセットアップ](#)」に進んでください。AWS の「サポート対象外リージョン」ページにリダイレクトされた場合は、新しいリージョンを選択する必要があります。
2. 適切なタブ (ライブストリーミング、IVS の場合は、ストリームチャット、IVS Chat の場合は) を選択し、リストされているリージョンのいずれかを選択します。後で必要になるので、どのリージョンを選択したのかをメモしておきます。

<https://aws.amazon.com/> の [My Account] (アカウント) をクリックして、いつでもアカウントの AWS のアクティビティを表示し、アカウントを管理することができます。

ステップ 2: ルートユーザーと管理ユーザーのセットアップ

AWS アカウントにサインアップすると、AWS アカウントのルートユーザーが作成されます。ルートユーザーには、アカウントのすべての AWS サービスとリソースへのアクセス権があります。セキュリティのベストプラクティスとして、[管理ユーザーに管理アクセスを割り当て、ルートユーザーアクセスが必要なタスク](#)を実行する場合にのみ、ルートユーザーを使用してください。

AWS アカウントのルートユーザーを保護する

1. IAM Identity Center に管理ユーザーとしてサインインするには、IAM Identity Center ユーザーの作成時に E メールアドレスに送信されたサインイン URL を使用します。IAM Identity Center ユーザーを使用してサインインする方法については、「[AWS サインインユーザーガイド](#)」の「AWS アクセスポータルにサインインする」を参照してください。

ルートユーザーを使用してログインする方法については、「AWS サインインユーザーガイド」の「[ルートユーザーとしてサインインする](#)」を参照してください。

2. ルートユーザーの多要素認証 (MFA) を有効にします。

手順については、「IAM ユーザーガイド」の「[AWS アカウントのルートユーザーの仮想 MFA デバイスを有効にする \(コンソール\)](#)」を参照してください。

管理ユーザーを作成する

日常的なタスクにルートユーザーを使用しないように、管理ユーザーを作成します。

- 日常的な管理タスクでは、AWS IAM アイデンティティセンター (AWS Single Sign-On の後継) の管理ユーザーに、管理アクセスを割り当てます。手順については、AWS IAM アイデンティティセンター (AWS Single Sign-On の後継) ユーザーガイドの「[はじめに](#)」を参照してください。
- IAM Identity Center に管理ユーザーとしてサインインするには、IAM Identity Center ユーザーの作成時に E メールアドレスに送信されたサインイン URL を使用します。IAM Identity Center ユーザーを使用してサインインする方法については、「[AWS サインインユーザーガイド](#)」の「AWS アクセスポータルにサインインする」を参照してください。

ステップ 3: IAM アクセス許可の設定

次に、基本的なアクセス許可セット (Amazon IVS チャンネルの作成、ストリーミング情報の取得、auto-record-to-S3 など) をユーザーに付与する AWS Identity and Access Management (IAM) ポリシーを作成し、そのポリシーをユーザーに割り当てる必要があります。[新しいユーザー](#)の作成時にアクセス許可を追加するか、あるいは[既存のユーザー](#)にアクセス許可を追加することができます。両方の手順を以下に示します。

詳細 (IAM ユーザーとポリシーについて、ポリシーをユーザーにアタッチする方法、Amazon IVS を使用してユーザーのアクションを制限する方法など) については、以下を参照してください。

- IAM ユーザーガイドの [IAM ユーザーの作成](#)
- IAMに関する [Amazon IVS のセキュリティ](#) と「IVSのマネージドポリシー」に関する情報。
- S3 への録画機能の場合: Amazon IVS ユーザーガイドの「[サービスにリンクされたロールの使用](#)」および「[Amazon S3 への自動録画](#)」

Amazon IVS 用の既存の AWS マネージドポリシーを使用するか、ユーザー、グループ、またはロールのセットに付与する権限をカスタマイズする新しいポリシーを作成できます。両方のアプローチを以下にて説明します。

IVS のアクセス許可には既存のポリシーを使用してください。

ほとんどの場合、Amazon IVS には AWS マネージドポリシーを使用することになります。これらについては、「IVS Security」の「[IVS 用マネージドポリシー](#)」セクションで詳しく説明されています。

- IVSReadOnlyAccess AWS マネージドポリシーを使用して、アプリケーション開発者がすべての IVS Get および List API エンドポイント (低レイテンシーとリアルタイムストリーミングの両方) にアクセスできるようにします。

- IVSFullAccess AWS マネージドポリシーを使用して、アプリケーション開発者がすべての IVS API エンドポイント (低レイテンシーとリアルタイムストリーミングの両方) にアクセスできるようにします。

オプション: Amazon IVS アクセス許可の新しいポリシーを作成する

以下の手順に従います。

1. AWS マネジメントコンソールにサインインして、IAM コンソールを開きます。 <https://console.aws.amazon.com/iam/>
2. ナビゲーションペインで、[ポリシー]、[ポリシーの作成] の順に選択します。[アクセス許可の指定] ウィンドウが開きます。
3. [アクセス許可の指定] ウィンドウで、[JSON] タブをクリックし、次の IVS ポリシーをコピーして [ポリシーエディタ] テキスト領域に貼り付けます。(このポリシーにはすべての Amazon IVS アクションが含まれるわけではありません。エンドポイントのアクセス許可は必要に応じて追加/削除 (許可/拒否) できます。IVS エンドポイントの詳細については「[IVS 低レイテンシーストリーミング API リファレンス](#)」を参照してください。)

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ivs:CreateChannel",
        "ivs:CreateRecordingConfiguration",
        "ivs:GetChannel",
        "ivs:GetRecordingConfiguration",
        "ivs:GetStream",
        "ivs:GetStreamKey",
        "ivs:GetStreamSession",
        "ivs:ListChannels",
        "ivs:ListRecordingConfigurations",
        "ivs:ListStreamKeys",
        "ivs:ListStreams",
        "ivs:ListStreamSessions"
      ],
      "Resource": "*"
    }
  ],
  {
```

```

    "Effect": "Allow",
    "Action": [
      "cloudwatch:DescribeAlarms",
      "cloudwatch:GetMetricData",
      "s3:CreateBucket",
      "s3:GetBucketLocation",
      "s3:ListAllMyBuckets",
      "servicequotas:ListAWSDefaultServiceQuotas",
      "servicequotas:ListRequestedServiceQuotaChangeHistoryByQuota",
      "servicequotas:ListServiceQuotas",
      "servicequotas:ListServices",
      "servicequotas:ListTagsForResource"
    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "iam:AttachRolePolicy",
      "iam:CreateServiceLinkedRole",
      "iam:PutRolePolicy"
    ],
    "Resource":
      "arn:aws:iam::*:role/aws-service-role/ivs.amazonaws.com/
      AWSServiceRoleForIVSRecordToS3*"
  }
]
}

```

4. [アクセス許可の指定] ウィンドウを開いたまま、[次へ] を選択します (ウィンドウの一番下までスクロールすると表示されます)。[レビューと作成] ウィンドウが開きます。
5. [レビューと作成] ウィンドウで、ポリシーにポリシー名を入力します。オプションで説明を追加します。ユーザーを作成する時に必要になるので、ポリシーの名前を書きとめておきます (下記を参照してください)。ページの最下部で、[Create policy] (ポリシーの作成) を選択します。
6. IAM コンソールウィンドウが表示され、新しいポリシーが作成されたことを確認するバナーが表示されます。

新しいユーザーを作成し、アクセス許可を付与する

IAM ユーザーアクセスキー

IAM アクセスキーは、アクセスキー ID とシークレットアクセスキーで構成されます。これは AWS へのプログラムによるリクエストの署名に使用されます。アクセスキーがない場合は、AWS マネジメントコンソールから作成できます。ベストプラクティスとして、ルートユーザーのアクセスキーは作成しないでください。

シークレットアクセスキーを表示またはダウンロードできるのは、アクセスキーを作成するときのみです。後で回復することはできません。ただ、アクセスキーはいつでも新しく作成できます。必要な IAM アクションを実行するためのアクセス許可が必要です。

アクセスキーは、常に安全に保管してください。(例え Amazon からの問い合わせであっても) 第三者と共有しないでください。詳細については、「IAM ユーザーガイド」の「[IAM ユーザーのアクセスキーの管理](#)」を参照してください。

手順

以下の手順に従います。

1. ナビゲーションペインで [ユーザー]、[ユーザーの作成] の順に選択します。[ユーザーの詳細を指定] ウィンドウが開きます。
2. [ユーザーの詳細を指定] ウィンドウで次の手順を実行します。
 - a. [ユーザーの詳細] で、作成する新しいユーザーの名前を入力します。
 - b. [AWS マネジメントコンソールへのユーザーアクセスを提供] を選択します。
 - c. プロンプトが表示されたら、[IAM ユーザーを作成します] を選択します。
 - d. [コンソールパスワード] で、[自動生成パスワード] を選択します。
 - e. [ユーザーは次回サインイン時に新しいパスワードを作成する必要があります] を選択します。
 - f. [次へ] をクリックします。[アクセス許可の設定] ウィンドウが開きます。
3. [アクセス許可の設定] で、[ポリシーを直接アタッチする] を選択します。[アクセス許可ポリシー] ウィンドウが開きます。
4. 検索ボックスに、IVS ポリシー名 (AWS マネージドポリシーまたは以前に作成したカスタムポリシー) を入力します。見つかったら、チェックボックスをオンにして、ポリシーを選択します。
5. ページの最下部で、[次へ] を選択します。[レビューと作成] ウィンドウが開きます。
6. [レビューと作成] ウィンドウで、ユーザーのすべての詳細が正しいことを確認してから、ウィンドウ最下部にある [ユーザーの作成] を選択します。

7. [パスワードの取得] ウィンドウが開き、コンソールサインインの詳細が表示されます。後で参照できるように、この情報を保存しておきます。完了したら、[ユーザーリストに戻る] を選択します。

既存のユーザーへのアクセス許可を追加する

以下の手順に従います。

1. AWS マネジメントコンソールにサインインして、IAM コンソールを開きます。 <https://console.aws.amazon.com/iam/>
2. ナビゲーションペインで、[Users (ユーザー)] を選択し、更新する既存のユーザー名を選択します。(名前をクリックして選択します。選択ボックスはチェックしないでください。)
3. 概要ページの[アクセス許可] タブで、[アクセス許可の追加] を選択します。[アクセス許可の追加] ウィンドウが開きます。
4. [既存のポリシーを直接アタッチ] を選択します。[アクセス許可ポリシー] ウィンドウが開きます。
5. 検索ボックスに、IVS ポリシー名 (AWS マネージドポリシーまたは以前に作成したカスタムポリシー) を入力します。ポリシーが見つかったら、チェックボックスをオンにして、ポリシーを選択します。
6. ページの最下部で、[次へ] を選択します。[レビュー] ウィンドウが開きます。
7. [レビュー] ウィンドウの下部にある [アクセス許可の追加] を選択します。
8. [Summary] (概要) ページで、IVS ポリシーが追加されたことを確認します。

ステップ 4: 任意の録画によるチャンネルの作成

Amazon IVS チャンネルには、ライブストリームに関連する設定情報が保存されます。まず、チャンネルを作成し、チャンネルのストリームキーを使用してチャンネルに動画を投稿し、ライブストリームを開始します。

チャンネル作成の一環として、以下の項目が割り当てられます。

- 取り込みサーバーは、ストリームを受信する特定の Amazon IVS コンポーネントと、取り込みプロトコル (RTMPS または RTMP) を識別します。
- チャンネルを作成し、ストリーミングを許可する際に、Amazon IVS はストリームキーを割り当てます。すべてのユーザーがチャンネルにストリーミングできるため、ストリームキーは機密情報として扱ってください。

- 再生 URL は、特定のチャンネルの再生を開始するエンドポイントを識別します。このエンドポイントはグローバルに使用できます。このエンドポイントは、Amazon IVS グローバルコンテンツ配信ネットワークから、視聴者が動画をストリーミングするのに最適な場所を自動的に選択します。(Amazon IVS は再生用のカスタムドメインをサポートしていないことに注意してください。自分のドメインで再生 URL をプロキシしないでください。ドメインが機能せず、問題が発生します。)

Amazon IVS コンソールまたは AWS CLI を介して、録画の有無にかかわらずチャンネルを作成できます。チャンネルの作成と録画については、以下で説明します。チャンネルの作成と録画については、以下で説明します。

Amazon S3 への自動録画

チャンネルの録画を有効にするオプションもあります。S3 への自動録画機能が有効になっている場合、チャンネル上のすべてのストリームが録画され、所有する Amazon S3 ストレージバケットに保存されます。その後、この録画はオンデマンド再生が可能です。

このセットアップは拡張オプションです。デフォルトでは、チャンネルの作成時に録画は無効になっています。

録画用のチャンネルを設定する前に、録画設定を作成する必要があります。これは、チャンネルの録画されたストリームが保存される Amazon S3 の場所を指定するリソースです。コンソールまたは CLI を使用して、録画設定を作成および管理することができます。両方の手順を以下に示します。録画設定を作成した後、チャンネルの作成時 (以下で説明) または後で既存のチャンネルを更新することで、録画設定をチャンネルに関連付けます。(API では、[CreateChannel](#) 「」および「」を参照してください [UpdateChannel](#)。) 複数のチャンネルを同じ録画設定に関連付けることができます。どのチャンネルにも関連付けられていない録画設定は削除できます。

次の制約に注意が必要です。

- S3 バケットを所有している必要があります。つまり、録画するチャンネルを設定するアカウントは、録画を保存する S3 バケットを所有している必要があります。
- チャンネル、録画設定、および S3 の場所は、同じ AWS リージョン内にある必要があります。他のリージョンでチャンネルを作成して録画する場合は、それらのリージョンで録画設定と S3 バケットも設定する必要があります。

S3 バケットに録画するには、AWS 認証情報を使用した承認が必要です。IVS に必要なアクセス権を付与するには、AWS IAM [サービスにリンクされたロール](#) (SLR) が録画設定を作成するときに自動

的に作成されます。SLR は、特定のバケットだけに対して IVS 書き込み許可を与えるように制限されます。

録画を有効にした後で再度、無効にするには、[the section called “録画を無効にする方法”](#)を参照してください。

ストリーミングロケーションと AWS 間、または AWS 内のネットワークの問題により、ストリームの録画中にデータが失われる可能性があることに注意してください。このような場合、Amazon IVS は録画よりもライブストリームを優先します。冗長性のために、ストリーミングツールでローカルに録画してください。

録画したファイルのポスト処理や VOD 再生の設定方法など、詳しい情報は「[Amazon S3 への自動録画](#)」を参照してください。

コンソールでの手順

これらのステップは、初期チャンネルのセットアップ、Amazon S3 への自動録画のセットアップ (オプション)、最終的なチャンネル作成の 3 つのフェーズに分かれています。

初期チャンネルの設定

1. [Amazon IVS コンソール](#)を開きます。

([AWS マネジメントコンソール](#)から Amazon IVS コンソールにアクセスすることもできます)。

2. ナビゲーションバーから、[Select a Region (リージョンの選択)] ドロップダウンをクリックして、リージョンを選択します。新しいチャンネルがこのリージョンに作成されます。
3. [Get started (開始する)] ボックス (右上) で、[Create Channel (チャンネルの作成)] を選択します。
4. [Channel configuration (チャンネル設定)] で、[Default configuration (デフォルト設定)] を許可します。オプションで、[Channel name (チャンネル名)] を指定します。チャンネル名は一意ではありませんが、チャンネル ARN (Amazon リソースネーム) 以外のチャンネルを区別するのに役立ちます。

注: カスタム設定を使用して、チャンネルタイプや (RTMPS の代わりに) RTMP インジェストなど、デフォルト以外の特定の値を指定できます。カスタム仕様はここには記載されていません。

Create channel [Info](#)

A channel is a unique configuration for streams. It includes broadcast configuration details (a server URL and stream key) for streaming software/hardware, and a playback URL for playing the stream. Channel configuration may affect pricing. [Amazon IVS Pricing](#) 

▶ How Amazon Interactive Video Service works

Setup

Channel name

Maximum length: 128 characters. May include numbers, letters, underscores (_) and hyphens (-).

Channel configuration

Default configuration
Use the default video latency and configuration, optimized for live interactions.

Custom configuration
Specify your own channel type and video latency configuration.

Channel type [Info](#)

Standard (broadcast and deliver live video up to 1080p Full HD, with transcoding and 1080p pass-through)

Video latency [Info](#)

Low (best for low-latency interactions with viewers)

Playback authorization [Info](#)

Disabled

Insecure ingest [Info](#)

Disabled

Restrict playback [Info](#)

Playback restriction policy [Info](#)

Restrict playback by country and origin.

Enable playback restriction

5. Amazon S3 に自動録画する場合は、以下の [Amazon S3 への自動録画の設定 \(オプション\)](#) に進みます。それ以外の場合は、スキップして直接[最終チャンネルの作成](#)に進んでください。

Amazon S3 への自動録画の設定 (オプション)

次のステップに従って、新しいチャンネルの作成中に録画を有効にします。

1. [チャンネルの作成] ページの [ストリームの記録と保存] で、[自動録画の有効化] を選択します。追加のフィールドが表示され、既存の [Recording configuration (録画設定)] を選択できます。新規の録画設定を作成することもできます。

Record and store streams [Info](#)

Auto-record to S3 [Info](#)
For improved redundancy, always record locally via your streaming tool.

Enable automatic recording

Recording configuration

 **Associated costs**
There are four cost components to consider when enabling record to S3: storage, request and data retrieval, data transfer, and data management. [Estimate data use.](#)

Tags [Info](#)
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

2. [Create recording configuration (録画設定の作成)] を選択します。新しいウィンドウが開きます。オプションを使用して、Amazon S3 バケットを作成し、新しい録画設定にアタッチします。

Create recording configuration ✕

Recording configuration name – *optional*

recording-configuration-1

Maximum length: 128 characters. May include numbers, letters, underscores (`_`) and hyphens (`-`).

Recording configuration

- Default configuration**
Use the default settings for auto-recording video and thumbnails.

- Custom configuration**
Specify your own video and thumbnail recording options.

Recorded renditions

All renditions

Thumbnail recording

Record at an interval - 60s

Target thumbnail resolution

Source (same resolution as input stream)

Thumbnail storage

Store thumbnails sequentially

Merge fragmented streams

Disabled

Storage

- Create a new Amazon S3 bucket
- Select an existing Amazon S3 bucket

Bucket name

ivs-stream-archive

The bucket name must be unique and must not contain spaces or uppercase letters. [See rules for bucket naming](#).

3. 次のフィールドに入力します。
 - a. 必要に応じて録画設定名を入力します。
 - b. [チャンネルの設定] で、[デフォルト設定] を有効にします。注意: カスタム構成は、記録されたレンディションや断片化されたストリームのマージなど、デフォルト以外の特定の値を指定する際に使用できません。カスタム仕様はここには記載されていません。
 - c. Bucket Name (バケット名)を入力します。
4. [Create] (作成) を選択して、一意の ARN を持つ新規の録画設定リソースを作成します。通常、録画設定の作成は数秒ですが、最大で 20 秒かかることがあります。録画設定が作成されると、[Create channel (チャンネルの作成)] ウィンドウに戻ります。[ストリーミングの録画と保存] エリアには、新しい記録設定が表示され、[状態] はアクティブで、作成した S3 バケット (ストレージ) が表示されます。

Record and store streams [Info](#)

Auto-record to S3 [Info](#)

For improved redundancy, always record locally via your streaming tool.

Enable automatic recording

Recording configuration

configuration-1 ▼



Create recording configuration

State

Active

Storage

s3-bucket-name [🔗](#)

Recording prefix [Info](#)

s3://ivs-r2s3-ivsstoragebucket-1kem14abgbit8/ivs/v1/298083573632/<attached_channel_id>/

Recorded renditions

All renditions

Merge fragmented streams

Disabled

Thumbnail recording

At 60-second intervals

Thumbnail storage

Store thumbnails sequentially

Thumbnail resolution

Source (same resolution as input stream)



Associated costs

There are four cost components to consider when enabling record to S3: storage, request and data retrieval, data transfer, and data management. [Estimate data use.](#)

▶ Tags [Info](#)

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Cancel

Create channel

最終チャンネルの作成

1. [Create channel (チャンネルの作成)] を選択し、一意の ARN で新しいチャンネルを作成します。新しいチャンネルのチャンネル詳細ページが表示されます。このページは開いたままにします。(注: 録画を有効にしなかった場合は、S3 への自動録画は、無効に設定されます。画面には、録画設定セクションが表示されません)。

channel-live Info

[Edit](#) [Delete](#)

General configuration

Channel name channel-live	Channel type Standard	Video latency Ultra-low
Playback authorization Disabled	Auto-record to S3 Enabled	ARN arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh

▶ Live stream

Stream configuration Info

[Reset stream key](#)

Stream key [Show](#)

Ingest server
 rtmps://a1b2c3d4e5f6.global-contribute.live-video.net:443/app/

▶ Other ingest options

Playback configuration Info

Playback URL
 https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8

Recording configuration Info

[Manage](#)

Recording configuration configuration-1	Storage s3-bucket-name	Recording prefix s3://qazwsxedcrfv-deleteme-1/ivs/v1/1234567891012/a1b2c3d4e5f6/
--	---	---

Stream sessions Info

< 1 >

Stream ID	Start time	Duration
-----------	------------	----------

No past streams to display
No past streams

Tags Info

[Manage tags](#)

2. 重要:

- [Stream configuration (ストリームの設定)] エリアで、取り込みサーバーおよびストリームキーを書き留めます。次のステップでこれらを使用し、ストリーミングを設定します。
- [Playback configuration (再生設定)] エリアで、再生 URL を書き留めます。後でそれを使用し、ストリームを再生します。

CLI の手順

AWS CLI を使用してチャンネルを作成することは詳細オプションであり、最初に CLI をダウンロードしてマシンに設定する必要があります。詳細については、[AWS コマンドラインインターフェイスのユーザーガイド](#)を参照してください。

チャンネル作成時に録画を有効 / 無効にするかどうかに応じて、次の 2 つの手順のいずれかに従います。

録画せずにチャンネルを作成する

1. create-channel コマンドを実行し、オプション名を渡します:

```
aws ivs create-channel --name test-channel
```

2. これにより、新しいチャンネルが返されます。

```
{
  "channel": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "authorized": false,
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
    "insecureIngest": false,
    "latencyMode": "LOW",
    "name": "channel-live",
    "playbackRestrictionPolicyArn": "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/abcdABCDefgh",
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
    "recordingConfigurationArn": "none",
    "tags": {},
    "type": "STANDARD"
  },
  "streamKey": {
```

```
"arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/g1H2I3j4k5L6",
"channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
"tags": {},
"value": "sk_us-west-2_abcdABCDefgh_567890abcdef"
}
}
```

3. **重要:** ingestEndpoint、streamKey value、および playbackUrl の点に注意してください。これらを使用して、ストリーミングと再生を設定します。

録画付きチャンネルの作成

前提条件 この手順を開始する前に Amazon S3 バケットを作成し、その ARN を書き留めます。[Amazon S3 の開始方法](#)を参照してください。S3 バケットは、録画設定を作成する同じリージョンに作成する必要があります。以下のステップ 1 の既知の問題を参照してください。

次に、以下のステップに従ってチャンネルを作成します。

1. create-recording-configuration コマンドを実行して、既存の Amazon S3 バケットの ARN を渡します。

```
aws ivs create-recording-configuration --name configuration-1 --destination-configuration s3={bucketName=test-bucket}
```

オプションで、thumbnail-configuration パラメータを渡してサムネイルの記録モードとサムネイル間隔を手動で設定します。

```
aws ivs create-recording-configuration --name configuration-1 --destination-configuration s3={bucketName=s3_bucket_name} --thumbnail-configuration recordingMode="INTERVAL",targetIntervalSeconds=60
```

必要に応じて recording-reconnect-window-seconds パラメータを渡して、断片化されたストリームの結合の機能を有効にします。

```
aws ivs create-recording-configuration --name configuration-1 --destination-configuration s3={bucketName=test-bucket} --recording-reconnect-window-seconds 60
```

既知の問題: us-east-1 リージョンでは、AWS CLI を使用して録画設定を作成すると、S3 バケットが別のリージョンにある場合でも成功が返されます。この場合、録画設定の state は、

(ACTIVEの代わりに) CREATE_FAILEDとなります。(他のリージョンでは、バケットが別のリージョンにある場合、CLI は正しく失敗を返します)。

回避方法: S3 バケットが録画設定と同じリージョンにあることを確認してください。S3 バケットとして別のリージョンに録画設定を作成する場合は、その録画設定を削除し、正しいリージョンから S3 バケットで新しい録画設定を作成します。

2. これにより、一意の ARN を持つ新しい録画設定が返されます。録画設定の状態がCREATINGの場合は、作成中であることを示しています。

```
{
  "recordingConfiguration": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/
mhndauNa01te",
    "name": "configuration-1",
    "destinationConfiguration": {
      "s3": {
        "bucketName": "s3_bucket_name"
      }
    },
    "recordingReconnectWindowSeconds": 60,
    "state": "CREATING",
    "tags": {},
    "thumbnailConfiguration": {
      "recordingMode": "INTERVAL",
      "targetIntervalSeconds": 60
    }
  }
}
```

3. 通常、録画設定の作成は数秒ですが、最大で 20 秒かかることがあります。録画設定が作成されたことを確認するには、get-recording-configuration コマンドを実行します。

```
aws ivs get-recording-configuration --arn "arn:aws:ivs:us-
west-2:123456789012:recording-configuration/mhndauNa01te"
```

4. これにより、録画設定が作成されたことを示すレスポンスが返されます (stateはACTIVE)。

```
{
  "recordingConfiguration": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/
mhndauNa01te",
    "name": "configuration-1",
```

```

    "destinationConfiguration": {
      "s3": {
        "bucketName": "s3_bucket_name"
      }
    },
    "recordingReconnectWindowSeconds": 60,
    "state": "ACTIVE",
    "tags": {},
    "thumbnailConfiguration": {
      "recordingMode": "INTERVAL",
      "targetIntervalSeconds": 60
    }
  }
}

```

5. チャンネルを作成し、そのチャンネルでの録画を有効にするには、`create-channel` コマンドを実行し、録画設定の ARN を渡します。

```

aws ivs create-channel --name channel-live --recording-configuration-arn
  "arn:aws:ivs:us-west-2:123456789012:recording-configuration/mhndauNa01te"

```

または、既存のチャンネルでの録画を有効にするために、`update-channel` コマンドを実行し、録画設定の ARN を渡します。

```

aws ivs update-channel --arn "arn:aws:ivs:us-west-2:123456789012:channel/
  abcdABCDefgh" --recording-configuration-arn "arn:aws:ivs:us-
  west-2:123456789012:recording-configuration/mhndauNa01te"

```

6. これにより、録画が有効であることを示す、`recordingConfigurationArn` の「none」以外の値を持つチャンネルオブジェクトが返されます。(以下の応答は `create-channel` からのものです。 `update-channel` レスポンスには `streamKey` オブジェクトは含まれません)。

```

{
  "channel": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
    "authorized": false,
    "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
    "insecureIngest": false,
    "latencyMode": "LOW",
    "name": "channel-live",
    "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
  video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",

```

```
"recordingConfigurationArn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/mhndauNa01te",
  "tags": {},
  "type": "STANDARD"
},
"streamKey": {
  "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/g1H2I3j4k5L6",
  "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
  "tags": {},
  "value": "sk_us-west-2_abcdABCDefgh_567890abcdef"
}
}
```

7. 重要: ingestEndpoint、streamKey value、および playbackUrl の点に注意してください。これらを使用して、ストリーミングと再生を設定します。

ステップ 5: ストリーミングソフトウェアの設定

以下の方法で Amazon IVS にストリーミング (低レイテンシー) できます。

- RTMPS をサポートする、ネイティブの [IVS Broadcast SDK](#)。特に本稼働シナリオでは、こちらをお勧めします。
- [Amazon IVS コンソール](#) – ストリームのテストに適しています。
- その他のストリーミングソフトウェア – [Open Broadcast Software \(OBS\)](#) と [FFmpeg](#) の 2 つの例について以下で説明します。これらは、Amazon IVS 同様に RTMPS および RTMP をサポートしています。RTMPS は、暗号化された TLS ストリームを使用して、高セキュリティを実現します。RTMPS または RTMP をサポートする任意のストリーミングソフトウェアを使用できます。

キーエンコーダの設定は、キーフレーム間隔 (2 秒) と解像度/ビットレート/フレームレート (相互に関連する) です。エンコーダ設定の詳細については、次を参照してください。

- Amazon IVS ユーザーガイドの「[ストリーミング設定](#)」を参照してください。
- ブログ記事: [Setting Up for Streaming with Amazon Interactive Video Service](#)

注意:

- Amazon IVS ストリームの最大継続時間は 48 時間です。その後、ストリームは終了し、ストリーミングセッションは切断されます。(自動または手動で) 正常に再接続すると、新しいストリームが開始されます。
- エンコーダーが (一時的なネットワークの問題などの原因で) データの送信を停止した場合、Amazon IVS は 30 秒間待機します。この間にブロードキャスターのデータが受信されない場合、Amazon IVS の接続が解除されます。

Amazon IVS Broadcast SDK によるストリーミング

iOS または Android アプリケーションからブロードキャストするには、Amazon IVS Broadcast SDK を使用できます。ブロードキャスト SDK は Amazon IVS アーキテクチャを活用し、Amazon IVS と共に継続的な改善と新しい機能を見ていきます。ネイティブのモバイルブロードキャスト SDK として、アプリケーションおよびユーザーがアプリケーションにアクセスするデバイスに対するパフォーマンスへの影響を最小限に抑えるように設計されています。

以下からブロードキャストするには..。	以下を使用できます..。	メモ
Android または iOS アプリケーション	Amazon IVS Android または iOS Broadcast SDK	ネイティブのモバイルブロードキャスト SDK として、アプリケーションおよびユーザーがアプリケーションにアクセスするデバイスに対するパフォーマンスへの影響を最小限に抑えるように設計されています。
ウェブ環境	Amazon IVS Web Broadcast SDK	Web Broadcast SDK として、Amazon IVS Web Broadcast SDK を使用すると、WebRTC を使用してウェブ環境からブロードキャストできます。クロスブラウザとクロスプラットフォームをサポートします。

詳細については、「[IVS Broadcast SDK](#)」を参照してください。

Amazon IVS コンソールを使用したストリーミング

1. [Amazon IVS コンソール](#)を開きます。

([AWS マネジメントコンソール](#)から Amazon IVS コンソールにアクセスすることもできます。)

2. ナビゲーションペインで、[チャンネル] を選択します。(ナビゲーションペインが折りたたまれている場合は、ハンバーガーアイコンを選択して展開します)
3. ブロードキャスト先のチャンネルを選択し、その詳細ページに移動します。
4. [ブロードキャスト] タブを選択します。(これらのタブは、[一般的な設定] セクションの下にあります)
5. カメラとマイクへの IVS コンソールアクセスを許可するように求められます。これらの権限を許可してください。
6. [ブロードキャスト] タブの下部にあるドロップダウンボックスを使用して、マイクとカメラの入力デバイスを選択します。
7. ストリーミングを開始するには、[ブロードキャストを開始] を選択します。
8. ライブストリームを視聴するには、[再生] タブに移動します。

注: ストリームを開始した後、[再生] タブで視聴できるようになるまでに、短い遅延 (通常は 30 秒未満) が発生することが予想されます。

この機能を使うと、複数のチャンネルに同時にブロードキャストできます。

注: コンソールからのストリーミングはリソースを消費するため、ライブ動画入力コストが発生します。詳細については、「IVS 料金表」の「[ライブ動画入力にかかるコスト](#)」を参照してください。

OBS Studio によるストリーミング

([OBS Studio](#)) は、録画およびライブストリーミングに適した無料のオープンソースソフトウェアです。OBS Studio は、リアルタイムのソースとデバイスのキャプチャ、シーンの構成、エンコーディング、記録、ストリーミングを提供します。

OBS Studio を起動して実行するには、次のステップに従います。

1. ソフトウェアをダウンロードしてインストールします: <https://obsproject.com/download>。
2. OBS Studio を初めてロードするときに表示される、OBS Studio の自動設定ウィザードを実行します。ステップに従い、デフォルト値を受け入れます。
3. [ストリーム情報] で、[サービス] ドロップダウンから [カスタム...] を選択し、[サーバー (取り込みサーバー)] と [ストリームキー] を入力します。

Amazon IVS コンソールでチャンネルを作成した場合:

- OBS で入力するサーバーは、コンソールの取り込みサーバーです。

```
rtmps://a1b2c3d4e5f6.global-contribute.live-video.net:443/app/
```

- OBS で入力するストリームキーは、コンソールのストリームキーです。

```
sk_us-west-2_abcd1234efgh5678ijkl
```

AWS CLI でチャンネルを作成した場合:

- OBS に入力するサーバーは、以下の 4 つの組み合わせです。
 - 取り込みプロトコル: `rtmps://` (注: 安全でない取り込みストリーミング用のチャンネルを作成した場合は `rtmp://` を指定してください)
 - CLI レスポンスからの `IngestEndpoint`:

```
a1b2c3d4e5f6.global-contribute.live-video.net
```

- ポート: 443
- パス: `/app/`

完全なエントリは次のとおりです。

```
rtmps://a1b2c3d4e5f6.global-contribute.live-video.net:443/app/
```

- OBS で入力するストリームキーは、CLI レスポンスの `StreamKey` 値です。

```
sk_us-west-2_abcd1234efgh5678ijkl
```

4. 動画出力の解像度とビットレートについては、「Amazon IVS ストリーミング設定」の「[チャンネルタイプ](#)」を参照してください。OBS ウィザードで選択したいいずれかの値が Amazon IVS で許可されている値を超える場合は、Amazon IVS への接続に失敗しないように手動で値を調整する必要があります。ウィザードの完了後、次の操作を行います。
 - a. 動画の解像度を調整するには、[Settings > Video > Output (Scaled) Resolution] (設定 > 動画 > 出力 (スケーリング) 解像度) を使用します。
 - b. 動画のビットレートを調整するには、[Settings > Output > Streaming > Video Bitrate] (設定 > 出力 > ストリーミング > 動画ビットレート) を使用します。
5. ストリームの安定性を高め、ビューア再生時のバッファリングを回避するために、2 秒間のキーフレーム間隔をお勧めします。ウィザードが完了したら、[Settings > Output > Output Mode] (設定 > 出力 > 出力モード) に進み、[Advanced] (アドバンスト) を選択し、[Streaming] (ストリーミング) タブで [Keyframe Interval] (キーフレーム間隔) が 2 になっていることを確認します。
6. OBS Studio のメインウィンドウで、[Start Streaming (ストリーミングの開始)] を選択します。

OBS Studio によるストリーミングの詳細については、[OBS Studio Quickstart](#)を参照してください。

OBS の設定は後で手動で変更できます。

1. [Settings (設定) > Stream (ストリーム)] を選択します。
2. ドロップダウンから、[Custom (カスタム)] を選択します。
3. サーバーおよび/またはストリームキーに貼り付けます。

ウィザードはいつでも再実行できます。[Tools (ツール) > Auto-Configuration Wizard (自動構成ウィザード)] を選択してください。

オプションで、ライブストリームをSettings (設定) > General (全般)にローカルに保存することで、後で使用できます。前述のように、ブロードキャストと AWS 間または AWS 内のネットワークの問題により、ストリームの録画中にデータが失われる可能性があります。このような場合、Amazon IVS は録画よりもライブストリームを優先します。ストリーミングツールでローカルに録画することで、冗長性が得られます。

OBS Studio のアップデートを定期的にチェックして、最新バージョンに更新することをお勧めします。(例えば、「Failed to connect to server (サーバーへの接続に失敗しました)」というエラーが表示された場合は、RTMPS をサポートしていない古いバージョンの OBS Studio を使用している可能性があります。)

FFmpeg による録画動画のストリーミング

以下のステップに従ってください。

1. FFmpeg をダウンロードしてインストールします。<https://www.ffmpeg.org/download.html>。
2. \$VIDEO_FILEPATHをストリーミングする MP4 動画の場所に設定します。

```
VIDEO_FILEPATH=/home/test/my_video.mp4
```

3. STREAM_KEY を StreamKey 値 に設定します。

```
STREAM_KEY=sk_us-west-2_abcd1234efgh5678ijkl
```

4. (ingestEndpoint に INGEST_ENDPOINT を設定します (AWS CLI から)。

```
INGEST_ENDPOINT=a1b2c3d4e5f6.global-contribute.live-video.net
```

5. 次のターミナルコマンドでストリーミングを開始します (これはすべて 1 行です)。

```
ffmpeg -re -stream_loop -1 -i $VIDEO_FILEPATH -r 30 -c:v libx264 -pix_fmt yuv420p  
-profile:v main -preset veryfast -x264opts "nal-hrd=cbr:no-scenecut" -minrate  
3000 -maxrate 3000 -g 60 -c:a aac -b:a 160k -ac 2 -ar 44100 -f flv rtmps://  
$INGEST_ENDPOINT:443/app/$STREAM_KEY
```

上記のコマンドは一例です。プロダクションストリーミングの場合は、必要に応じてパラメータを調整します。

ステップ 6: ライブストリームの表示

ライブストリームを表示するには

1. [Amazon IVS コンソール](#)を開きます。

([AWS マネジメントコンソール](#)から Amazon IVS コンソールにアクセスすることもできます。)

2. ナビゲーションペインで、[Live channels] (ライブチャンネル) を選択します。(ナビゲーションペインが折りたたまれている場合は、まずハンバーガーアイコンを選択して開きます。)

3. ストリームを表示するチャンネルを選択し、そのチャンネルの詳細ページに移動します。

ライブストリームは、詳細ページのライブストリームセクションで再生されています。

注: コンソールからの再生はリソースを消費するため、ライブ動画出力コストが発生します。詳細については、「IVS 料金表」の「[ライブ動画出力にかかるコスト](#)」を参照してください。

注: ストリーミング開始後、コンソールでストリームを表示できるようになるまでに少し時間がかかります (通常は 30 秒以内)。

ステップ 7: Service-Quota 制限の確認 (オプション)

すべてのアカウントは、同時視聴者と同時ブロードキャストの数に制限があります。制限が適切であることを確認し、必要に応じて引き上げをリクエストします (特に大規模なストリーミングイベントを計画している場合)。詳細については、「[サービスクォータ \(低レイテンシーストリーミング\)](#)」を参照してください。

ステップ 8: 望ましくないコンテンツや視聴者を防ぐ (推奨)

悪意のあるユーザーは、プラットフォームで望ましくないコンテンツ (プロスポーツなど) を再ストリーミングしたり、許可なくプラットフォームのストリームを別のウェブサイトに埋め込もうとしたりする可能性があります。この種のストリーミングは、ビジネスに価値を付加することなく、アプリケーションが提供するライブストリーミングビデオの量とそれに関連するコストを劇的に増加させる可能性があります。Amazon IVS は、アクティブなストリームを停止するコントロールを提供するだけでなく、この種の動作を検出して防止するのに役立つリソースも提供します。[「望ましくないコンテンツと視聴者」](#)を参照してください。

再生を特定のオリジンや国に制限するには、再生制限ポリシーを使用します。

コンソールの手順 (プレイバック制限ポリシー)

1. 再生制限ポリシーを作成する

- [Amazon IVS コンソールを開きます](#)。左側のナビゲーションペインで、再生セキュリティ > 再生制限ポリシー を選択します。
- [Create policy] (ポリシーを作成) を選択します。
- 必要に応じて、ポリシーに名前を付けます。
- オプションで、厳密なオリジン強制を切り替えます (以下の注意を参照)。
- 許可される国と許可されるオリジンを指定します。
- [Create policy] (ポリシーを作成) を選択します。

2. このポリシーを新規または既存のチャンネルにアタッチする

- 新しいチャンネルを作成するか、既存のチャンネルを編集します。
- 「再生の制限」セクション (「チャンネルの作成」または「チャンネルの更新」ウィンドウ) で、「再生制限を有効にする」を選択します。
- 再生制限ポリシーのドロップダウンリストから、ステップ 1 で作成したポリシーを選択します。
- チャンネルの作成 (新しいチャンネルの場合) または保存 (既存のチャンネルを更新する場合) を選択します。

厳格なオリジン強制に関する注意：これは、許可されたオリジンで指定されたオリジン制限を強化するために使用できるオプション設定です。デフォルトでは、オリジン制限はマルチバリエーションプレイリストにのみ適用されます。厳密なオリジン強制が有効になっている場合、サーバーは、リクエスト元のオリジンがすべての再生リクエスト (マルチバリエーションプレイリスト、バリ

アントプレイリスト、セグメントを含む) のトークンと一致するという要件を適用します。つまり、すべてのクライアント (ブラウザ以外のクライアントを含む) は、各リクエストで有効なオリジンリクエストヘッダーを提供する必要があります。setOrigin メソッドを使用して、IVS iOS および Android プレイヤー SDK でヘッダーを設定します。iOS Safari を除くウェブブラウザでは自動的に設定されます。iOS Safari の場合、オリジンリクエストヘッダーが確実に送信されるように、動画要素に `crossorigin="anonymous"` を追加する必要があります。例えば、`<video crossorigin="anonymous"></video>` などです。

IP アドレスと国間のマッピングに関する注意： IVS は、サードパーティーのデータベースを使用してユーザーの場所を決定します。IP アドレスと国とのマッピングの正確さは、リージョンによって異なります。最近のテストによれば、全体的な正確性は 99.8% です。IVS がユーザーの場所を特定できない場合、IVS はユーザーが要求したコンテンツを提供します。

CLI の手順 (Playback 制限ポリシー)

1. 再生制限ポリシーを作成します。以下はその例です。 `allowed-countries` および `allowed-origins` フィールドについては、以下の例の値を実際の値に置き換えるか、ユースケースに応じて一方または両方のフィールドを削除します。

```
aws ivs create-playback-restriction-policy --name test-playback-restriction-policy
--enable-strict-origin-enforcement --allowed-countries "US","JP" --allowed-origins
"https://example1.com","https://*.example2.com"
```

これにより、新しい再生制限ポリシーが返されます。そのフィールドについては、「IVS 低レイテンシーストリーミング API リファレンス [PlaybackRestrictionPolicy](#)」の「」を参照してください。

2. 新しいポリシーをチャンネルにアタッチします。既存のチャンネルについては、 を実行し、前のステップで作成した再生制限ポリシーの ARN を `update-channel` 渡します。

```
aws ivs update-channel --arn "arn:aws:ivs:us-west-2:123456789012:channel/
abcdABCDefgh" --playback-restriction-policy-arn "arn:aws:ivs:us-
west-2:123456789012:playback-restriction-policy/abcdABCDefgh"
```

新しいチャンネルの場合は、チャンネルの作成時に `--playback-restriction-policy-arn` ステートメントを含めます。 [???](#)

録画を無効にする方法

既存のチャンネルで Amazon S3 録画を無効にするには

- コンソール — 該当するチャンネルの詳細ページにある [Record and store (記録と保存)] ストリームセクションで、[Disabled (無効)] を選択し、次に [Save Channel (チャンネルの保存)] を選択します。これにより、録画設定とチャンネルとの関連付けが解除され、そのチャンネル上のストリームは録画されなくなります。
- CLI — `update-channel` コマンドを実行し、録画設定の ARN を空の文字列として渡します。

```
aws ivs update-channel --arn "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh" --recording-configuration-arn ""
```

これにより、録画が無効であることを示す、`recordingConfigurationArn` の空の文字列を持つチャンネルオブジェクトが返されます。

Amazon IVS ストリームで複数ホストを有効にする

Amazon Interactive Video Service (IVS) により、開発者は複数のブロードキャスター (ホストとも呼ばれます) の動画とオーディオを結合するアプリケーションを構築できます。

ユースケースを以下に示します。

- ゲストスポット — ブロードキャスターは視聴者をブロードキャストに招待できます。これにより、カラオケや Q&A などのコラボレーションコンテンツへの扉が開かれます。
- バーサス (VS) モード — ブロードキャスター同士がマッチングして競い合います (歌唱コンクールなど)。
- グループブロードキャスト — 複数のスピーカーが大勢の視聴者の前で互いに会話できます。

ライブストリームに複数のブロードキャスターを追加するには、IVS リアルタイムストリーミングと IVS 低レイテンシーストリーミングの両方を使用する必要があります。IVS リアルタイムストリーミングはビデオストリームとオーディオストリームを組み合わせるもので、低レイテンシーストリーミングは組み合わせたストリームを視聴者にブロードキャストします。

リアルタイムストリーミングは、ステージと呼ばれるリソースを提供します。これはブロードキャスター (ホスト) がリアルタイムでオーディオと動画を交換できる仮想スペースです。その後、ステージをチャンネルにブロードキャストしてより多くの視聴者にリーチしたり、視聴者を「ステージ」に連れて行って進行中の会話に貢献できるアプリケーションを構築したりできます。

IVS リアルタイムストリーミングの詳細については、以下を参照してください。

- [IVS リアルタイムストリーミングユーザーガイド](#)
 - IVS Broadcast SDK にはリアルタイム機能が組み込まれています。これらの SDK ([ウェブ](#)、[Android](#)、[iOS](#)) のガイド、特に「公開と購読」のセクションを参照してください。
- 「[IVS リアルタイムストリーミング API リファレンス](#)」

開始方法

コンソールでの手順

新しいステージとそのステージの参加者トークンを作成するには、次の手順に従います。

1. [Amazon IVS コンソール](#)を開きます。

([AWS マネジメントコンソール](#)から Amazon IVS コンソールにアクセスすることもできます。)

2. 左側のナビゲーションペインで [ステージ] を選択し、[ステージを作成] を選択します。[ステージを作成] ウィンドウが表示されます。

Amazon IVS > Video > Stages > Create stage

Create stage [Info](#)

A stage allows participants to send and receive video and audio with others in real time. You can broadcast a stage to a channel, allowing viewers to see and hear stage participants without needing to join the stage directly. [Learn more](#)

▶ **How Amazon IVS stages work**

Setup

Stage name – optional

Maximum length: 128 characters. May include numbers, letters, underscores (_) and hyphens (-).

▶ **Tags** [Info](#)

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Cancel **Create stage**

3. 必要に応じて、[ステージ名] を入力します。[ステージを作成] を選択してステージを作成します。新しいステージのステージ詳細ページが表示されます。
4. [参加者トークンを作成] を選択します。
5. [参加者トークンを作成] ダイアログで、ユーザー ID を入力し、[参加者トークンを作成] を選択します。トークンは [参加者トークン] テーブルの上部に表示されます。トークンをコピーするには、[トークンをコピー] のアイコン (参加者トークンの左側) をクリックします。

ステージのブロードキャスト: クライアントサイトとサーバーサイドのコンポジションの対比

開発者がステージを IVS チャンネルにブロードキャストする場合には、以下の 2 種類の選択肢があります。

- クライアントサイドコンポジションでは、ホストはステージに接続し、他のホストから動画をダウンロードし、それらを 1 つのストリームに結合して、結合済みのストリームを IVS チャンネルにブロードキャストします。この方法では、アプリ開発者が Mixer API を使用してコンポジションの外観を制御できるなど、レイアウトの柔軟性が非常に高くなります。ただし、クライアントサイドコンポジションでは、コンポジションを作成するためにクライアント CPU リソースがより多く必要であり、コンポジションのブロードキャストにもより多くの帯域幅が要求されます。また、ステージをブロードキャストするホストにネットワークの問題があると、視聴者に向けたライブストリームにも影響が出ることがあります。

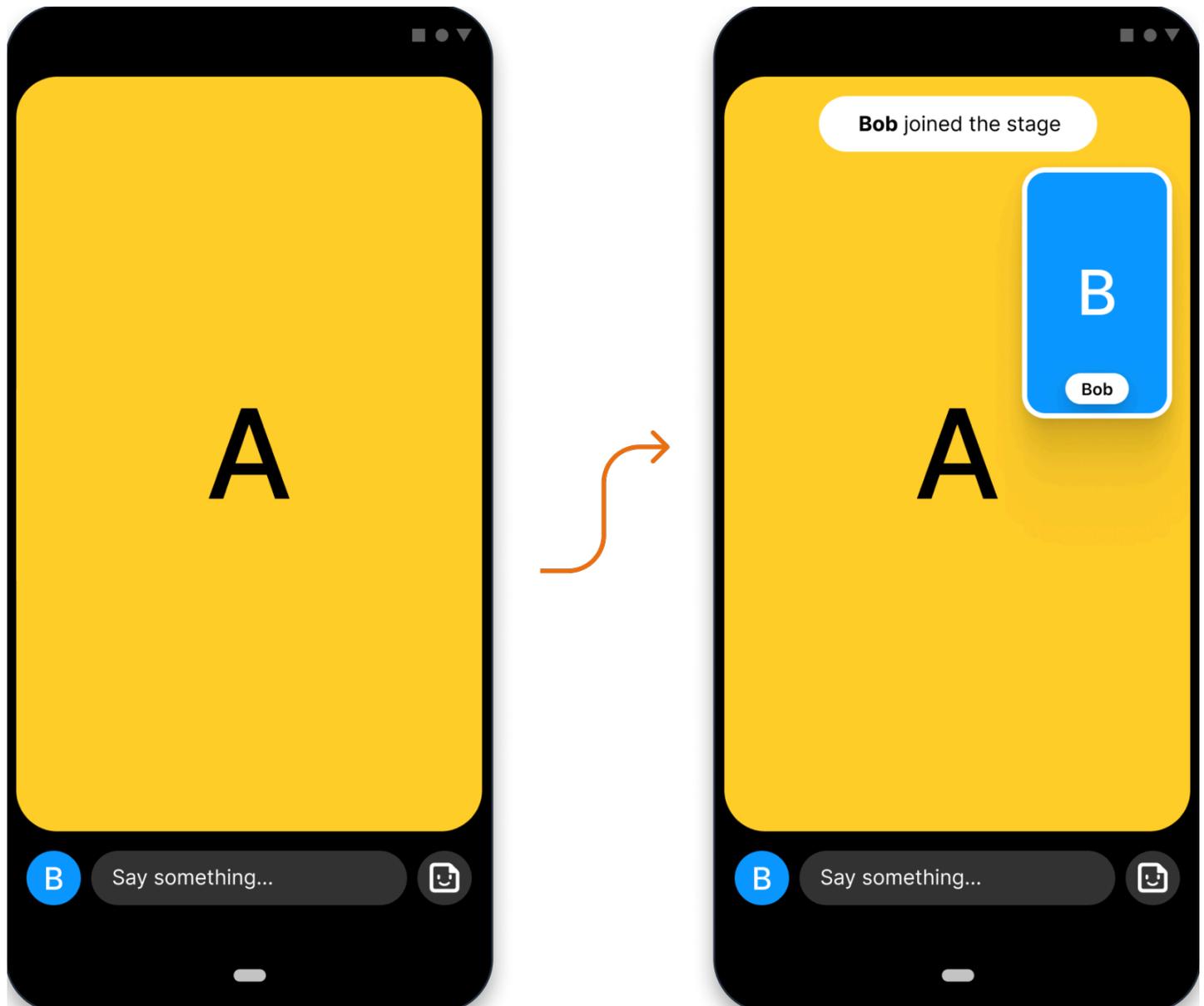
オーバーレイを組み込んだり、サーバーサイドコンポジションと互換性のない要素をカスタマイズしたりするなど、ブロードキャストのコンテンツを高度にパーソナライズする必要がある場合には、クライアントサイドコンポジションが推奨されます。

- サーバーサイドコンポジションでは、クライアントは、IVS ステージのコンポジションとブロードキャストをクラウドサービスにオフロードします。サーバーサイドコンポジションとチャンネルへの RTMP のブロードキャストは、ステージのホームリージョンにある IVS のコントロールプレーンエンドポイントを介して呼び出されます。数多くの利点を備えたサーバーサイドコンポジションは、効率的で信頼性の高いライブストリーミングを求めるユーザーにとって、魅力的な選択肢です。
- 軽減されたクライアント負荷 – サーバーサイドコンポジションでは、音声と動画ソースを組み合わせる作業が、個別のクライアントデバイスからサーバー側に移転されます。サーバーサイドコンポジションにより、ビューを合成して IVS に送信するクライアントデバイスは、CPU とネットワークリソースを使用しなくても良くなります。
- レジリエンス – コンポジションプロセスをサーバー上で一元化することで、ブロードキャストをより堅牢にできます。パブリッシャーのデバイスに技術的な制限がかかっていたり、ネットワークの変動があつたりしても、サーバーはこれに適応し、すべての視聴者にスムーズなストリームを提供できます。
- 帯域幅の効率性 – サーバーがコンポジションを処理するため、ステージパブリッシャーは、IVS チャンネルに動画をブロードキャストする帯域幅を余分に消費する必要がありません。

詳しくは、「IVS リアルタイムストリーミングユーザーガイド」の「[サーバーサイドコンポジション](#)」を参照してください。

デモ

シナリオ: Alice (A) は Amazon IVS チャンネルにブロードキャストしていて、Bob (B) をゲストとしてステージに招待したいと考えています。(実際のブロードキャストでは、A と B はアリスとボブの画像になります。)



1. ステージの作成

Amazon IVS ステージ API を使用した [CreateStage](#) リクエストは次のとおりです。

```
POST /CreateStage HTTP/1.1
Content-type: application/json
{
  "name": "string",
  "participantTokenConfigurations": [
    {
      "userId": "9529828585",
      "attributes": {"displayName": "Alice"}
    },
    {
      "userId": "4875935192",
      "attributes": {"displayName": "Bob"}
    }
  ]
}
```

ここでやっているように、ステージを作成するときに参加者トークンを事前に作成できません。[CreateParticipantToken](#) を呼び出すことで、既存のステージのトークンを作成することもできます。参加者ごとに、カスタム `userId` と `attributes` のセットを渡すことができます。(重要: `attributes` および `userId` リクエストフィールドはすべてのステージ参加者に公開されます。これらを個人を特定する情報、機密情報、または機密情報には使用しないでください。)

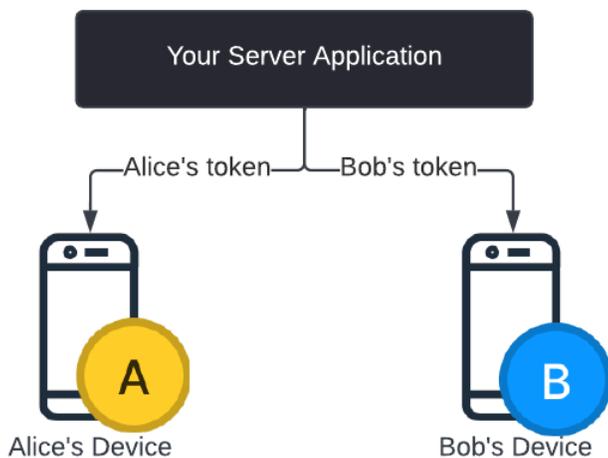
上記のリクエストに対するネットワークの応答は次のとおりです。

```
HTTP/1.1 200
Content-type: application/json
{
  "stage": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh",
    "name": "alice-stage"
  },
  "participantTokens": [
    {
      "participantId": "e94e506e-f7...",
      "token": "eyJhbGciOiJI...",
      "userId": "9529828585",
      "attributes": {"displayName": "Alice"},
      "expirationTime": number
    }
  ]
}
```

```
  },  
  {  
    "participantId": "b5c6a79a-6e...",  
    "token": "eyJhbGciOiJ...",  
    "userId": "4875935192",  
    "attributes": {"displayName": "Bob"},  
    "expirationTime": number  
  }  
]  
}
```

2. 参加者トークンの配布

これで、クライアントは Alice (A) と Bob (B) のトークンを手に入れました。デフォルトでは、トークンは 1 時間有効です。オプションで、ステージの作成時にカスタム duration を渡すことができます。トークンはステージに参加するために使用できます。

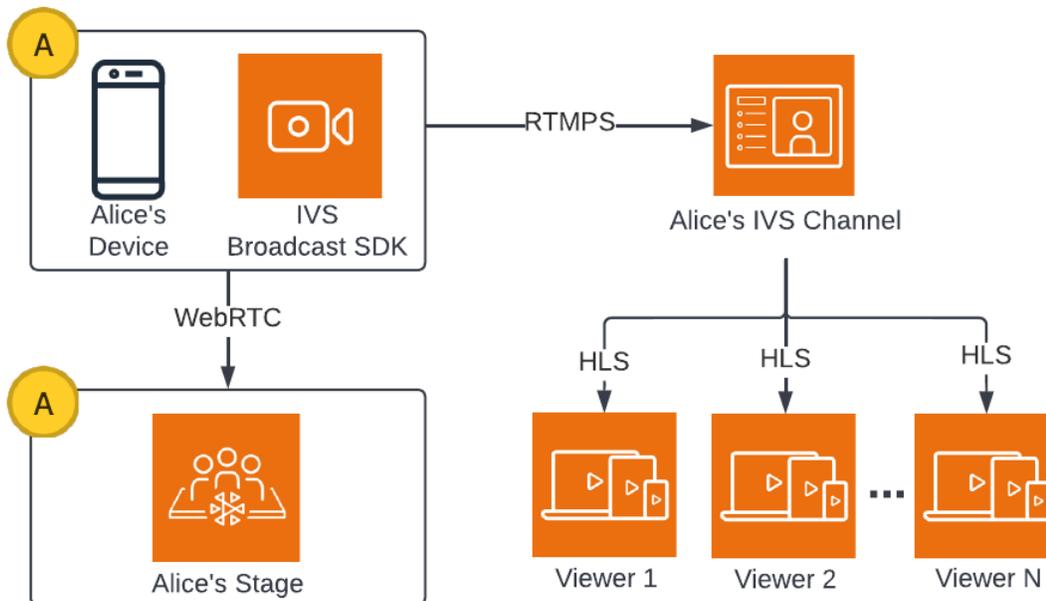


サーバーから各クライアントにトークンを配布する方法が必要になります (WebSocket チャンネル経由など)。この機能は提供していません。

3. ステージに参加する

参加者は Android または iOS の Amazon IVS Broadcast SDK を介してステージに参加できます。各参加者の動画の品質を設定できます。ここでは、アリスが最初にステージに参加する様子を示します。

アーキテクチャの概要は次のとおりです。



そして、これがステージに参加するための Android コードサンプルです。以下のコードスニペットは Alice のデバイスで動作します。join() の呼び出しでは、Alice がステージに参加します。上の図は、このコード実行の結果を示しており、Alice はステージに参加して公開中です (ステップ 1 で始めたチャンネルへのブロードキャストに加えて)。

```
// Create streams with the front camera and first microphone.
var deviceDiscovery = DeviceDiscovery(context)
var devices : List<Device> = deviceDiscovery.listLocalDevices()
var publishStreams = ArrayList<LocalStageStream>()

// Configure video quality if desired
var videoConfiguration = StageVideoConfiguration()

// Create front camera stream
var frontCamera = devices.find { it.descriptor.type ==
    Device.Descriptor.DeviceType.Camera && it.descriptor.position ==
    Device.Descriptor.Position.FRONT }
var cameraStream = ImageLocalStageStream(frontCamera, videoConfiguration)
publishStreams.add(cameraStream)

// Create first microphone stream
var microphone = devices.find { it.descriptor.type ==
    Device.Descriptor.DeviceType.Microphone }
var microphoneStream = AudioLocalStageStream(microphone)
publishStreams.add(microphoneStream)
```

```
// A basic Stage.Strategy implementation that indicates the user always wants to
// publish and subscribe to other participants.
// Provides the front camera and first microphone as publish streams.

override fun shouldPublishFromParticipant(stage: Stage, participantInfo:
    ParticipantInfo) : Boolean {
    return true
}

override fun shouldSubscribeToParticipant(stage: Stage, participantInfo:
    ParticipantInfo) : Stage.SubscribeType {
    return Stage.SubscribeType.AUDIO_VIDEO
}

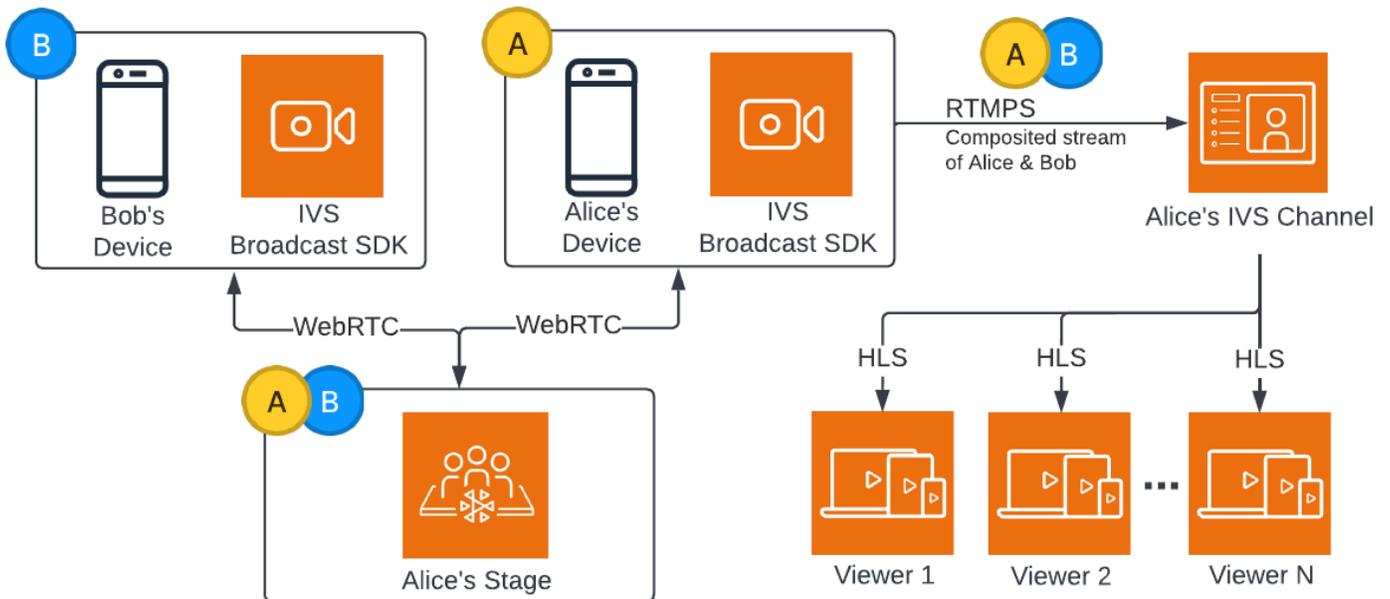
override fun stageStreamsToPublishForParticipant(stage: Stage, participantInfo:
    ParticipantInfo): List<LocalStageStream> {
    return publishStreams
}

// Create Stage using the strategy and join
var stage = Stage(context, token, strategy)

try {
    stage.join()
} catch (exception: BroadcastException) {
    // handle join exception
}
```

4. ステージをブロードキャストする

クライアントサイドコンポジション



ステージをブロードキャストするための Android コードサンプルを次に示します。

```
var broadcastSession = BroadcastSession(context, broadcastListener, configuration,
    null)

// StageRenderer interface method to be notified when remote streams are available
override fun onStreamsAdded(stage: Stage, participantInfo: ParticipantInfo, streams:
    List<StageStream>) {

    var id = participantInfo.participantId

    // Create mixer slot for remote participant
    var slot = BroadcastConfiguration.Mixer.Slot.with { s ->
        s.name = id
        // Set other properties as desired
        ...
        s
    }

    broadcastSession.mixer.addSlot(slot)

    // Attach remote stream devices, bind to mixer slot
```

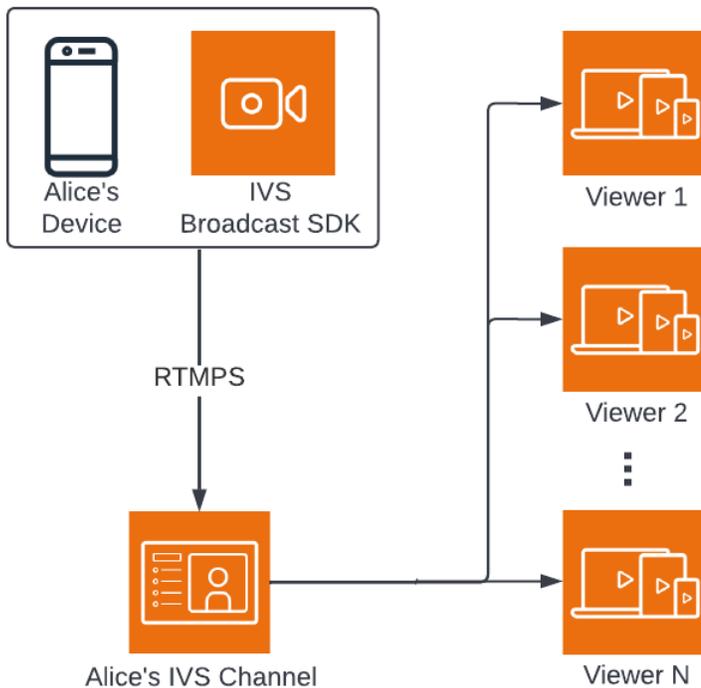
```
streams.forEach { stream ->
    broadcastSession.attachDevice(stream.getDevice())
    broadcastSession.mixer.bind(stream.getDevice(), id)
}
}

// Start broadcasting
try {
    broadcastSession.start(IVS_RTMP_URL, IVS_STREAM_KEY)
} catch (exception: BroadcastException) {
    // handle exception
}
```

Android と iOS の Amazon IVS Broadcast SDK には、動的な UI を簡単に構築できるように、参加者のステータス (例: `onStreamsAdded` と `onStreamsRemoved`) によってトリガーされるコールバックがあります。これはコードサンプルの最初の部分に示されており、Bob のビデオとオーディオが視聴可能になると、Alice に `onStreamsAdded` コールバックで通知されます。

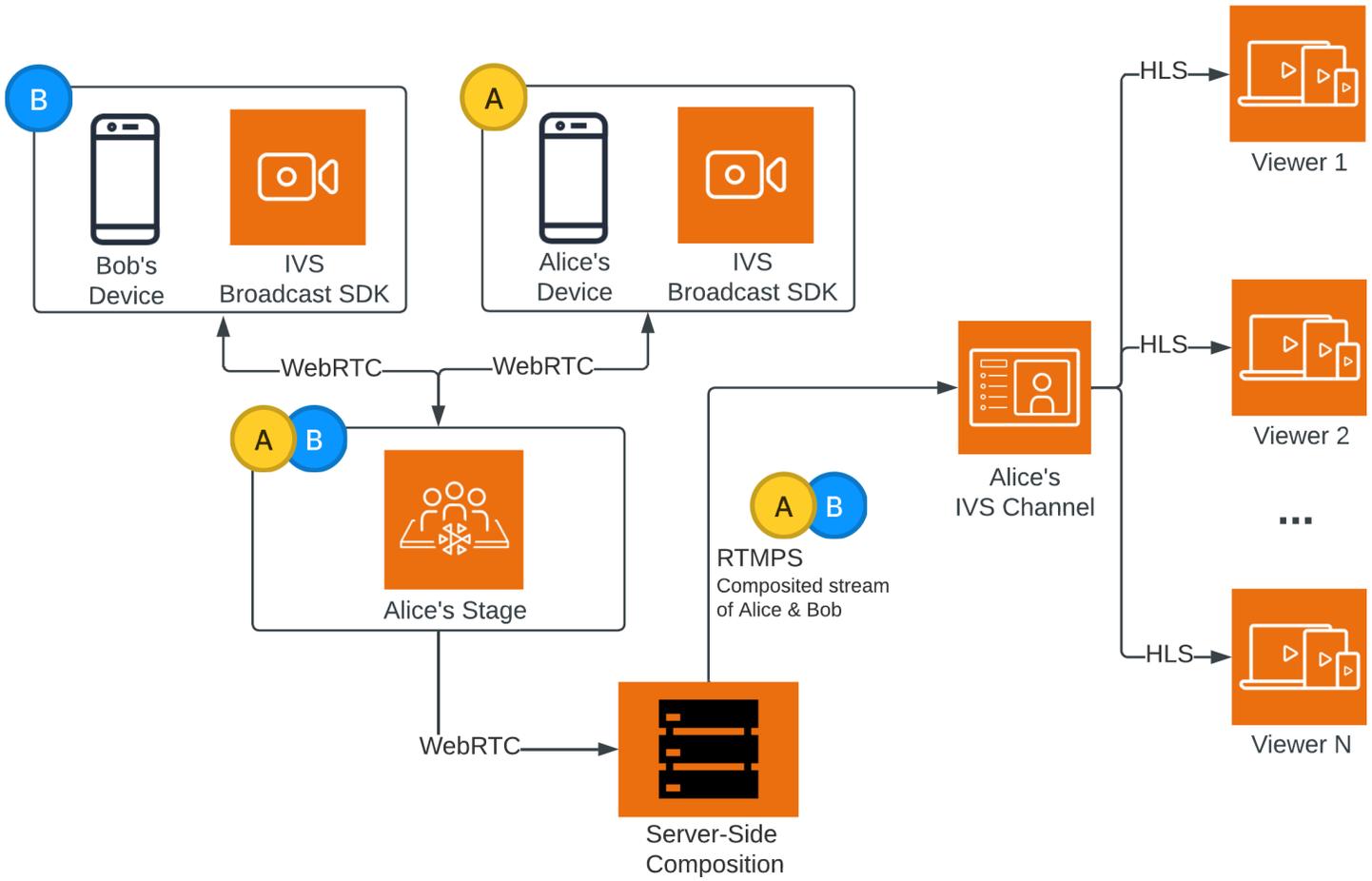
その後、Alice は Bob のビデオとオーディオをミキサーに追加して、チャンネルの幅広い視聴者向けに RTMP ブロードキャストに含めることができます。これはコードサンプルの残りの部分に示されています。

現在、Alice は Amazon IVS Android ブロードキャスト SDK を介して複数の視聴者にブロードキャストしています。これはアーキテクチャ的には次のようになります。



サーバーサイドコンポジション

比較のため、[サーバーサイドコンポジション](#)の仕組みを以下に示します。(詳細については、「IVS リアルタイムストリーミングユーザーガイド」の「[サーバーサイドコンポジション](#)」を参照してください。)



Amazon IVS Low-Latency Streaming のモニタリング

Amazon CloudWatch を使用して、Amazon Interactive Video Service (IVS) のリソースをモニタリングできます。CloudWatch は Amazon IVS から raw データを収集し、リアルタイムに近い読み取り可能なメトリクスに加工します。これらの統計は 15 か月間保持されるため、ウェブアプリケーションやサービスの動作に関する履歴情報を取得できます。特定のしきい値にアラームを設定し、これらのしきい値に達したときに通知を送信したりアクションを実行したりできます。詳細については、[CloudWatch ユーザーガイド](#)を参照してください。

メトリクスのタイムスタンプは、メトリクスデータが蓄積される期間の開始を表します。例: 1 分間のLiveDeliveredTimeメトリクスの合計 (01:02:00 に 300 秒)。これは、01:02:00 ~ 01:02:59 の 1 分間に、5 分間分の動画が視聴者に配信されたことを意味します。

高解像度として指定されたメトリクスの場合、最初のデータポイントはストリーミング開始から数秒後に表示されます。メトリクスリクエストを行うときは、期間として 5 秒を指定することをお勧めします。(Amazon CloudWatch ユーザーガイドの [解像度](#) を参照してください。) その他のメトリクスについては、参照先のタイムスタンプから 1 分以内にデータが送信されます。

高解像度メトリクスは、時間の経過とともにロールアップされます。メトリクスの経過とともに解像度は実質的に低下します。スケジュールは次のとおりです。

- 1 秒のメトリクスは 3 時間使用できます。
- 60 秒のメトリクスは 15 日間使用できます。
- 5 分間のメトリクスは 63 日間使用できます。
- 1 時間のメトリクスは、455 日 (15 か月) 間使用できます。

データ保持に関する最新情報については、[Amazon CloudWatch のよくある質問](#)の「保持期間」を検索してください。

前提条件

- Stream Health API および CloudWatch メトリクスと対話するのに十分な IAM アクセス許可を持つ AWS アカウントが必要です。具体的な手順については、「[IVS 低レイテンシーストリーミングの開始](#)」を参照してください。
- チャンネルを作成し、ストリームを開始する必要があります。[IVS 低レイテンシーストリーミングユーザーガイド](#)に関連情報があります。

- チャンネルの作成手順については、「Amazon IVS 低レイテンシーストリーミングの開始」の「[チャンネルの作成](#)」を参照してください。
- ストリームの開始手順については、「Amazon IVS 低レイテンシーストリーミングの開始」の「[ストリーミングソフトウェアの設定](#)」を参照してください。
- エンコーダー設定の詳細については、[Amazon IVS ストリーミング設定](#)を参照してください。

ストリームセッションデータへのアクセス

`listStreamSessions` エンドポイントを使用して、チャンネルが最大 60 日間保持しているストリームのリストにアクセスできます。このリストには、ライブストリームセッション (空の `endTime` で示される) が含まれる場合があります。

`getStreamSession` エンドポイントを使用して、特定のストリームのセッションデータを取得できます。`streamId` パラメータを指定しない場合、エンドポイントは最新のセッションを返します。さらに、エンドポイントを定期的に呼び出して、ストリームの最新のイベント (最新 500 個まで) を取得することもできます。

コンソールでの手順

1. [Amazon IVS コンソール](#)を開きます。

([AWS マネジメントコンソール](#)から Amazon IVS コンソールにアクセスすることもできます。)

2. ナビゲーションペインで、[Channels] (チャンネル) を選択します。(ナビゲーションペインが折りたたまれている場合は、まずハンバーガーアイコンを選択して開きます。)
3. チャンネルを選択して、その詳細ページに移動します。
4. [Stream sessions] (ストリームセッション) セクションが表示されるまでページを下にスクロールします。
5. アクセスするセッションのストリーム ID を選択して、Amazon CloudWatch 高解像度メトリクスのチャートを含む、セッションの詳細を表示します。

または、1 つ以上のチャンネルが既にライブ配信中の場合:

1. [Amazon IVS コンソール](#)を開きます。
2. ナビゲーションペインで、[Live channels] (ライブチャンネル) を選択します。(ナビゲーションペインが折りたたまれている場合は、まずハンバーガーアイコンを選択して開きます。)
3. 一覧からライブチャンネルを選択すると、分割ビュー内のセッション詳細が表示されます。

AWS SDK での手順

AWS SDK によるストリームセッションデータへのアクセスは詳細オプションであり、まず SDK をダウンロードしてアプリケーション上で設定する必要があります。以下は、JavaScript による AWS SDK の手順です。

前提条件: 以下のコードサンプルを使用するには、AWS JavaScript SDK をアプリケーションにロードする必要があります。詳細については、[Getting started with the AWS SDK for JavaScript](#) を参照してください。

```
// This first call lists up to 50 stream sessions for a given channel.
const AWS = require("aws-sdk");
const REGION = 'us-west-2';
let channelArn = USE_YOUR_CHANNEL_ARN_HERE;

AWS.config.getCredentials(function(err) {
  if (err) console.log(err.stack);
  // credentials not loaded
  else {
    console.log("Access key:", AWS.config.credentials.accessKeyId);
  }
});

AWS.config.update({region: REGION});
var ivs = new AWS.IVS();

// List Stream Sessions
async function listSessions(arn) {
  const result = await ivs.listStreamSessions({"channelArn": arn}).promise();
  console.log(result.streamSessions);
}
listSessions(channelArn);

// Get Stream Session
async function getSession(arn, id) {
  const result = await ivs.getSession({"channelArn": arn, "streamId":
id}).promise();
  console.log(result);

  // This function polls every 3 seconds and prints the latest IVS stream events.
  setInterval(function(){
    console.log(result.streamSession.truncatedEvents);
  }, 3000);
```

```
}  
getSession(channelArn);
```

CLI の手順

AWS CLI によるストリームセッションデータへのアクセスは詳細オプションであり、まず CLI をダウンロードしてマシン上で設定する必要があります。詳細については、[AWS Command Line Interface のユーザーガイド](#)を参照してください。

1. 次のように、ストリームセッションを一覧表示します。

```
aws ivs list-stream-sessions --channel-arn <arn>
```

2. 次のように、特定のストリームの streamId を使用して、そのストリームセッションデータを取得します。

```
aws ivs get-stream-session --channel-arn <arn> --stream-id <streamId>
```

get-stream-session 呼び出しのサンプルレスポンスを次に示します。

```
{  
  "streamSession": {  
    "startTime": "2021-10-22T00:03:57+00:00",  
    "streamId": "st-1FQzeLONMT9XTKI431eLSo1",  
    "truncatedEvents": [  
      {  
        "eventTime": "2021-10-22T00:09:30+00:00",  
        "name": "Session Ended",  
        "type": "IVS Stream State Change"  
      },  
      {  
        "eventTime": "2021-10-22T00:09:30+00:00",  
        "name": "Stream End",  
        "type": "IVS Stream State Change"  
      },  
      {  
        "eventTime": "2021-10-22T00:03:57+00:00",  
        "name": "Stream Start",  
        "type": "IVS Stream State Change"  
      },  
      {
```

```
        "eventTime": "2021-10-22T00:03:50+00:00",
        "name": "Session Created",
        "type": "IVS Stream State Change"
    }
],
"endTime": "2021-10-22T00:09:31+00:00",
"ingestConfiguration": {
    "audio": {
        "channels": 2,
        "codec": "mp4a.40.2",
        "sampleRate": 48000,
        "targetBitrate": 160000
    },
    "video": {
        "avcLevel": "4.0",
        "avcProfile": "Baseline",
        "codec": "avc1.42C028",
        "encoder": "obs-output module (libobs version 27.0.1)",
        "targetBitrate": 3500000,
        "targetFramerate": 30,
        "videoHeight": 1080,
        "videoWidth": 1920
    }
},
"channel": {
    "name": "",
    "ingestEndpoint": "3f234d592b38.global-contribute.live-video.net",
    "authorized": false,
    "latencyMode": "LOW",
    "recordingConfigurationArn": "",
    "type": "STANDARD",
    "playbackUrl": "https://3f234d592b38.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.991729659840.channel.dY7LsluQX1gV.m3u8",
    "arn": "arn:aws:ivs:us-west-2:991729659840:channel/dY7LsluQX1gV"
}
}
}
```

ヘルスによるストリームのフィルタリング

問題が発生しているストリームを簡単に見つけるために、`listStreams` を使用してライブストリームを「ヘルス」でフィルタリングできます。

コンソールでの手順

1. [Amazon IVS コンソール](#)を開きます。

([AWS マネジメントコンソール](#)から Amazon IVS コンソールにアクセスすることもできます。)

2. ナビゲーションペインで、[Live channels] (ライブチャンネル) を選択します。(ナビゲーションペインが折りたたまれている場合は、まずハンバーガーアイコンを選択して開きます。)
3. [Filter by health] (ヘルスによるフィルタリング) の検索フィールドを選択します。
4. ドロップダウンリストから [Health = STARVING] (ヘルス = スタベーション) を選択します。

フィルタリング後、チャンネルの詳細ページに移動し、チャンネルのライブストリームセッションを選択して、入力設定の詳細とストリームイベントにアクセスできます。

CLI の手順

AWS CLI の使用は詳細オプションであり、まず CLI をダウンロードしてマシン上で設定する必要があります。詳細については、[AWS Command Line Interface のユーザーガイド](#)を参照してください。

次のように、ヘルス (STARVING など) によってストリームをフィルタリングします。

```
aws ivs list-streams --filter-by health=STARVING
```

ConcurrentStreams の CloudWatch ヘルスディメンション

ConcurrentStreams を特定の Health によってフィルタリングできます。「[CloudWatch メトリクス: IVS 低レイテンシーストリーミング](#)」を参照してください。

CloudWatch メトリクスへのアクセス

Amazon CloudWatch は Amazon IVS から raw データを収集し、ほぼリアルタイムの読み取り可能なメトリクスに加工します。これらの統計は 15 か月間保持されるため、ウェブアプリケーションやサービスの動作に関する履歴情報を取得できます。特定のしきい値にアラームを設定し、これらのしきい値に達したときに通知を送信したりアクションを実行したりできます。詳細については、[CloudWatch ユーザーガイド](#)を参照してください。

CloudWatch メトリクスは時間の経過とともにロールアップされることに注意してください。メトリクスの経過とともに解像度は実質的に低下します。スケジュールは次のとおりです。

- 1 秒のメトリクスは 3 時間使用できます。
- 60 秒のメトリクスは 15 日間使用できます。
- 5 分間のメトリクスは 63 日間使用できます。
- 1 時間のメトリクスは、455 日 (15 か月) 間使用できます。

getMetricData を呼び出すときに、高解像度メトリクスに、1、5 (推奨)、10、30、または 60 秒の任意の倍数の期間を指定できます。

CloudWatch コンソールでの手順

1. CloudWatch コンソール (<https://console.aws.amazon.com/cloudwatch/>) を開きます。
2. サイドナビゲーションで、[メトリクス] ドロップダウンをクリックし、[すべてのメトリクス] を選択します。
3. [参照] タブで、左側のラベルなしのドロップダウンを使用して、チャンネルが作成された「ホーム」リージョンを選択します。リージョンの詳細については、「[グローバルソリューション、リージョナルコントロール](#)」を参照してください。対応するリージョンの一覧については、「[AWS 全般のリファレンス](#)」の Amazon IVS のページを参照してください。
4. [参照] タブの下部で [IVS] 名前空間を選択します。
5. 次のいずれかを行います:
 - a. 検索バーに、リソース ID (ARN の一部、arn::`ivs:channel/<resource id>`) を入力します。

[IVS] > [チャンネル別] の順に選択します。
 - b. AWS の名前空間に IVS が選択可能なサービスとして表示される場合、IVS を選択します。Amazon IVS を使用して、Amazon CloudWatch にメトリクスを送信している場合にリスト表示されます。(IVS がリスト表示されていない場合、Amazon IVS メトリクスはありません)。

次に、必要に応じてディメンショングループを選択します。使用可能なディメンションは、以下の「[CloudWatch メトリクス](#)」にリストされています。
6. グラフに追加するメトリクスを選択します。利用可能なメトリクスは、以下の「[CloudWatch メトリクス](#)」にリストされています。

ストリームセッションの詳細ページで [CloudWatch で表示] ボタンを選択して、ストリームセッションの CloudWatch グラフにアクセスすることもできます。

CLI の手順

AWS CLI を使用してメトリクスにアクセスすることもできます。そのためには、まず CLI をマシンにダウンロードして設定する必要があります。詳細については、[AWS コマンドラインインターフェイスのユーザーガイド](#)を参照してください。

次に、AWS CLI を使用して Amazon IVS Low-Latency Streaming メトリクスにアクセスするために、次の操作を実行します。

- コマンドプロンプトで、次のコマンドを実行します。

```
aws cloudwatch list-metrics --namespace AWS/IVS
```

詳細については、Amazon CloudWatch ユーザーガイドの「[Amazon CloudWatch メトリクスの使用](#)」を参照してください。

CloudWatch メトリクス: IVS 低レイテンシーストリーミング

Amazon IVS は、AWS/IVS の名前空間で以下のメトリクスを提供します。

メトリクス	ディメンション	説明
ConcurrentViews	なし	<p>すべてのライブチャンネルでの同時視聴回数。ビューとは、動画をアクティブにダウンロードまたは再生している固有の視聴セッションです。(より詳細な定義については、用語集を参照してください。) チャンネルがライブで集計に視聴がない場合、メトリクスの値は 0 です。ライブのチャンネルがない場合、メトリクスにはデータポイントがありません。</p> <p>単位: 個</p> <p>有効な統計: 平均、最大、最小 — 設定された間隔における同時視聴の (それぞれの) 平均数、最大数、最小数</p>

メトリクス	ディメンション	説明
ConcurrentViews	Channel	<p>チャンネル ARN ごとに ConcurrentViews をフィルターします。チャンネルがライブで視聴がない場合、メトリクスの値は 0 です。チャンネルがライブでない場合、メトリクスにはデータポイントがありません。</p> <p>このメトリクスは、ストリームではなくチャンネルのデータを提供します。特定のチャンネル上の特定のストリーミングセッションの同時視聴を表示するには、ストリームセッションの開始時刻と終了時刻間のチャンネルのConcurrentViews メトリクスを評価します。</p> <p>単位: 個</p> <p>有効な統計: 平均、最大、最小 — 設定された間隔における同時視聴の (それぞれの) 平均数、最大数、最小数</p>
ConcurrentStreams	なし	<p>ライブでストリーミングしているチャンネルの数。ライブのチャンネルがない場合、このメトリクスにはデータポイントがありません。</p> <p>単位: 個</p> <p>有効な統計: 平均、最大、最小 — 設定された間隔における同時視聴の (それぞれの) 平均数、最大数、または最小数。</p>

メトリクス	ディメンション	説明
ConcurrentStreams	Health	<p>チャンネルのヘルスごとに ConcurrentStreams をフィルターします。ライブのチャンネルがない場合、このメトリクスにはデータポイントがありません。</p> <p>単位: 個</p> <p>有効な統計: 平均、最大、最小 — 設定された間隔における、特定の Health の同時ストリームの平均数、最大数、または最小数 (それぞれ)。</p>
IngestAudioBitrate	Channel	<p>(高解像度メトリクス) ストリーミング時に Amazon IVS が受け取るオーディオデータの量。ビットレートが高いほど、利用可能なインターネット帯域幅を使用します。</p> <p>単位: ビット/秒</p> <p>有効な統計: 平均、最大、最小 — 設定された間隔でのオーディオ取り込みビットレートの (それぞれの) 平均数、最大数、または最小数。</p>
IngestFramerate	Channel	<p>(高解像度メトリクス) ストリーミング時に Amazon IVS がアニメーションフレームを受信する頻度。</p> <p>単位: フレーム/秒</p> <p>有効な統計: 平均、最大、最小 — 設定された間隔での取り込みフレームレートの (それぞれの) 平均数、最大数、または最小数。</p>

メトリクス	ディメンション	説明
IngestVideoBitrate	Channel	<p>(高解像度メトリクス) ストリーミング時に Amazon IVS が受け取る動画データの量。ビットレートが高いほど、利用可能なインターネット帯域幅を使用します。ビットレートを上げると動画の品質が向上しますが、特定のポイントまでしか向上しません。</p> <p>単位: ビット/秒</p> <p>有効な統計: 平均、最大、最小 — 設定された間隔での動画取り込みビットレートの (それぞれの) 平均数、最大数、または最小数。</p>
KeyframeInterval	Channel	<p>(高解像度メトリクス) 前のフレームとの違いだけでなく、フレーム全体が送信される動画ストリーム内のポイント。</p> <p>単位: 秒</p> <p>有効な統計: 平均、最大、最小 — 設定された間隔でのキーフレーム間隔の (それぞれの) 平均数、最大数、または最小数。</p>
LiveDeliveredTime	なし	<p>すべての視聴者に配信された動画のリアルタイム時間の合計。</p> <p>単位: 秒</p> <p>有効な統計: Sum</p>
LiveDeliveredTime	Channel	<p>チャンネルごとに LiveDeliveredTime をフィルターします。チャンネル値はチャンネルの resource-id であり、これは ARN の最後の部分です。</p> <p>単位: 秒</p> <p>有効な統計: Sum</p>

メトリクス	ディメンション	説明
LiveDeliveredTime	Channel, ViewerCountryCode	<p>チャンネルおよび視聴者の国コードごとに LiveDeliveredTime をフィルターします。チャンネル値はチャンネルの resource-id であり、これは ARN の最後の部分です。国の値は 2 文字の ISO 3166-1 国コードです。これにより、視聴者の国を特定できます。視聴者の国を特定できない場合は、UNKNOWNとして表示されます。</p> <p>単位: 秒</p> <p>有効な統計: Sum</p>
LiveInputTime	なし	<p>動画ストリームのリアルタイム時間。</p> <p>単位: 秒</p> <p>有効な統計: Sum</p>
LiveInputTime	Channel	<p>チャンネルごとに LiveInputTime をフィルターします。チャンネル値はチャンネルの resource-id であり、これは ARN の最後の部分です。</p> <p>単位: 秒</p> <p>有効な統計: Sum</p>
RecordedTime	なし	<p>録画された動画のリアルタイム時間。</p> <p>単位: 秒</p> <p>有効な統計: Sum</p>

メトリクス	ディメンション	説明
RecordedTime	Channel	<p>チャンネルごとに RecordedTime をフィルターします。チャンネル値はチャンネルの resource-id であり、これは ARN の最後の部分です。</p> <p>単位: 秒</p> <p>有効な統計: Sum</p>

IVS Broadcast SDK (低レイテンシーストリーミング)

Amazon Interactive Video Service (IVS) 低レイテンシー Broadcast SDK は、Amazon IVS を使用してアプリケーションを構築するデベロッパー向けのものです。この SDK は、Amazon IVS のアーキテクチャを活用するように設計されており、Amazon IVS と共に、継続的に改善され、新機能が提供されます。ネイティブのブロードキャスト SDK として、アプリケーションおよびユーザーがアプリケーションにアクセスするデバイスに対してパフォーマンスへの影響を最小限に抑えるように設計されています。

アプリケーションでは、Amazon IVS Broadcast SDK の主な機能を活用できます。

- **高品質ストリーミング** — ブロードキャスト SDK は、高品質のストリーミングをサポートします。カメラから動画をキャプチャし、最大 1080p の品質でエンコードし、高品質の表示エクスペリエンスを実現します。
- **自動ビットレート調整** — スマートフォンユーザーは移動するため、ブロードキャストの過程でネットワークの状況が変わることがあります。Amazon IVS Broadcast SDK は、変化するネットワーク状況に対応するために、動画のビットレートを自動的に調整します。
- **縦向きと横向きのサポート** — ユーザーがデバイスをどのように持っているかに関係なく、画像は正しい向きで適切に拡大縮小されて表示されます。Broadcast SDK は、ポートレートとランドスケープの両方のキャンバスサイズをサポートします。ユーザーが設定した向きからデバイスを回転させると、アスペクト比が自動的に調整されます。
- **セキュアなストリーミング** — ユーザーのブロードキャストは、TLS を使用して暗号化されるため、ストリームを安全に保つことができます。
- **外部オーディオデバイス** — Amazon IVS Broadcast SDK は、オーディオジャック、USB、および Bluetooth SCO 外部マイクをサポートしています。

プラットフォームの要件

ネイティブプラットフォーム

プラットフォーム	サポートされているバージョン
Android	5.0 (Lollipop) 以降

プラットフォーム	サポートされているバージョン
iOS	12 以降 アプリケーションにブロードキャストが不可欠な場合は、Apple App Store からアプリケーションを UIRequiredDeviceCapabilities を利用してダウンロードするための要件として Metal を指定します。

IVS は、少なくとも 4 つの iOS メジャーバージョンと 6 つの Android メジャーバージョンをサポートしています。現在のサポート対象バージョンは、これらの最小値よりも多い可能性があります。メジャーバージョンがサポートされなくなる場合は、少なくとも 3 か月前に SDK リリースノートでお客様にお知らせします。

デスクトップブラウザ

ブラウザ	サポートされているプラットフォーム	サポートされているバージョン
Chrome	Windows、macOS	2 つのメジャーバージョン (最新バージョンと 1 つ前のバージョン)
Firefox	Windows、macOS	2 つのメジャーバージョン (最新バージョンと 1 つ前のバージョン)
Edge	(Windows 8.1 以降)	2 つのメジャーバージョン (最新バージョンと 1 つ前のバージョン) Edge Legacy は除く
Safari	macOS	2 つのメジャーバージョン (最新バージョンと 1 つ前のバージョン)

モバイルブラウザ

ブラウザ	サポートされるバージョン
Chrome for iOS、Safari for iOS	2つのメジャーバージョン (最新バージョンと1つ前のバージョン)
iPadOS 向け Chrome、iPadOS 向け Safari	2つのメジャーバージョン (最新バージョンと1つ前のバージョン)
Chrome for Android	2つのメジャーバージョン (最新バージョンと1つ前のバージョン)

ウェブビュー

Web Broadcast SDK は、ウェブビューやウェブライク環境 (テレビ、家庭用ゲーム機など) をサポートしていません。モバイル実装については、「[Android](#)、[iOS](#) 向け低レイテンシーストリーミング Broadcast SDK ガイド」を参照してください。

必要なデバイスのアクセス

Broadcast SDK では、デバイス内蔵のカメラとマイクと、Bluetooth、USB、またはオーディオジャックを介して接続されているカメラとマイクにアクセスする必要があります。

サポート

ストリームにブロードキャストエラーやその他の問題が発生した場合は、ブロードキャスト API を介して再生セッションの一意の識別子を特定します。

この Amazon IVS Broadcast SDK の場合 :	以下を使用
Android	BroadcastSession の getSessionId 機能
iOS	IVSBroadcastSession の sessionId プロパティ

この Amazon IVS Broadcast SDK の場合 :	以下を使用
Web	getSessionId 関数

このブロードキャストセッション識別子を AWS サポートに伝えます。それにより、サポートはトラブルシューティングに役立つ情報を入手できます。

注: Broadcast SDK は継続的に改善されています。利用可能なバージョンと修正済みの問題については [Amazon IVS リリースノート](#) を参照してください。必要な場合、サポートに連絡する前にお使いのプレイヤーのバージョンを更新し、問題が解決するかどうか確認してください。

バージョンニング

Amazon IVS Broadcast SDK は、[セマンティックバージョンニング](#) を使用しています。

以下の解説は、次を前提としています。

- 最新リリースは 4.1.3。
- 1 つ前のメジャーバージョンの最新リリースは 3.2.4。
- バージョン 1.x の最新リリースは 1.5.6。

最新バージョンのマイナーリリースとして、下位互換性のある新機能が追加されています。この場合、次回の新機能のセットは、バージョン 4.2.0 として追加されます。

下位互換性のあるマイナーなバグ修正が、最新バージョンのパッチリリースとして追加されています。ここでは、次回のマイナーなバグ修正のセットは、バージョン 4.1.4 として追加されます。

下位互換性のあるメジャーなバグ修正は異なる方法で処理されます。これらはいくつかのバージョンに追加されています。

- 最新バージョンのパッチリリース。こちらは、バージョン 4.1.4 です。
- 1 つ前のマイナーバージョンのパッチリリース。こちらは、バージョン 3.2.5 です。
- 最新バージョン 1.x リリースのパッチリリース。こちらは、バージョン 1.5.7 です。

メジャーなバグ修正は、Amazon IVS 製品チームによって定義されています。典型的な例に、重要なセキュリティ更新のほか、お客様に必要な選別された修正があります。

注: 上記の例では、リリースされたバージョンの数字は、連番でインクリメントされています(4.1.3 → 4.1.4、など)。実際は、1つ以上のパッチ番号が内部に残り、リリースされないままになることもあります。そのため、リリースされたバージョンは 4.1.3 から (例えば) 4.1.6 に増えることもあります。

IVS Broadcast SDK: Web ガイド (低レイテンシーストリーミング)

IVS 低レイテンシーストリーミング Web Broadcast SDK は、Web 上でインタラクティブかつリアルタイムの体験を構築するためのツールをデベロッパーに提供します。

Web Broadcast SDK の最新バージョン : 1.8.0 ([リリースノート](#))

リファレンスドキュメント : Amazon IVS Web Broadcast SDK で使用できる最も重要な方法については、[「https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference」](https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference) を参照してください。SDK の最新バージョンが選択されていることを確認してください。

サンプルコード: SDK をすぐに使い始めるには、以下のサンプルの利用が適しています。

- [IVS チャンネルへの単一ブロードキャスト \(HTML および JavaScript \)](#)
- [IVS チャンネルへ画面共有した単一ブロードキャスト \(React ソースコード\)](#)

プラットフォーム要件: サポートされているプラットフォームのリストについては、[「Amazon IVS Broadcast SDK」](#) を参照してください。

開始方法

ライブラリのインストール

IVS は [reflect-metadata](#)BroadcastClient を利用して、グローバル reflect オブジェクトを拡張することに注意してください。これによって競合が発生することはありませんが、まれに望ましくない動作が発生する可能性があります。

スクリプトタグを使用する

Web Broadcast SDK は JavaScript ライブラリとして配布され、<https://web-broadcast.live-video.net/1.8.0/amazon-ivs-web-broadcast.js> で取得できます。

<script> タグを使用してロードすると、ライブラリは IVSBroadcastClient という名前のウィンドウスコープにグローバル変数を公開します。

NPM を使用する

npm パッケージをインストールするには:

```
npm install amazon-ivs-web-broadcast
```

これで、IVSBroadcastClient オブジェクトにアクセスして、Errors や BASIC_LANDSCAPE のような他のモジュールや定数を取り込むことができます。

```
import IVSBroadcastClient, {
  Errors,
  BASIC_LANDSCAPE
} from 'amazon-ivs-web-broadcast';
```

サンプル

すぐに使用を開始するには、以下の例を参照してください。

- [IVS チャンネルへの単一ブロードキャスト \(HTML および JavaScript \)](#)
- [IVS チャンネルへ画面共有した単一ブロードキャスト \(React ソースコード\)](#)

AmazonIVSBroadcastClient のインスタンスを作成する

ライブラリを使用するには、クライアントのインスタンスを作成する必要があります。そのためには、解像度やフレームレートなどの配信時の制約を指定する streamConfig パラメータとともに、IVSBroadcastClient で create メソッドを呼び出します。インジェストエンドポイントは、クライアントの作成時またはストリームの開始時に設定できます。

取り込みエンドポイントは AWS コンソールで見つけるか、CreateChannel エンドポイント (例: UNIQUE_ID.global-contribute.live-video.net) によって返されます。

```
const client = IVSBroadcastClient.create({
  // Enter the desired stream configuration
  streamConfig: IVSBroadcastClient.BASIC_LANDSCAPE,
  // Enter the ingest endpoint from the AWS console or CreateChannel API
  ingestEndpoint: 'UNIQUE_ID.global-contribute.live-video.net',
});
```

以下は一般的にサポートされているストリーム設定です。プリセットは、最大で 480p および 1.5 Mbps ビットレートの BASIC、最大で 1080p および 3.5 Mbps ビットレートの BASIC Full HD、最大

で 1080p および 8.5 Mbps ビットレートの STANDARD (または ADVANCED) です。必要に応じてビットレート、フレームレート、解像度をカスタマイズできます。詳細については、「」を参照してください [BroadcastClientConfig](#)。

```
IVSBroadcastClient.BASIC_LANDSCAPE;  
IVSBroadcastClient.BASIC_FULL_HD_LANDSCAPE;  
IVSBroadcastClient.STANDARD_LANDSCAPE;  
IVSBroadcastClient.BASIC_PORTRAIT;  
IVSBroadcastClient.BASIC_FULL_HD_PORTRAIT;  
IVSBroadcastClient.STANDARD_PORTRAIT;
```

npm パッケージを使用する場合は、これらを個別にインポートできます。

注: クライアント側の設定がバックエンドのチャンネルタイプと一致していることを確認してください。例えば、チャンネルタイプが STANDARD の場合、streamConfig は IVSBroadcastClient.STANDARD_* 値のいずれかに設定する必要があります。チャンネルタイプが ADVANCED の場合は、以下にあるとおりに手動で設定する必要があります (ADVANCED_HD を例として使用)。

```
const client = IVSBroadcastClient.create({  
  // Enter the custom stream configuration  
  streamConfig: {  
    maxResolution: {  
      width: 1080,  
      height: 1920,  
    },  
    maxFramerate: 30,  
    /**  
     * maxBitrate is measured in kbps  
     */  
    maxBitrate: 3500,  
  },  
  // Other configuration . . .  
});
```

必要なアクセス許可

アプリケーションは、ユーザーのカメラとマイクへのアクセス許可をリクエストする必要があります。また、HTTPS で提供される必要があります。(これは Amazon IVS に特有ではなく、カメラやマイクにアクセスが必要なすべてのウェブサイトが必要です。)

オーディオおよびビデオデバイス両方のアクセス許可をリクエストし、キャプチャする方法を示す関数の例を次に示します。

```
async function handlePermissions() {
  let permissions = {
    audio: false,
    video: false,
  };
  try {
    const stream = await navigator.mediaDevices.getUserMedia({ video: true, audio:
true });
    for (const track of stream.getTracks()) {
      track.stop();
    }
    permissions = { video: true, audio: true };
  } catch (err) {
    permissions = { video: false, audio: false };
    console.error(err.message);
  }
  // If we still don't have permissions after requesting them display the error
message
  if (!permissions.video) {
    console.error('Failed to get video permissions.');
```

詳細については、「[Permissions API](#)」および[MediaDevicesgetUserMedia「\(\)」](#)を参照してください。

ストリームプレビューを設定する

配信される内容をプレビューするには、SDK に `<canvas>` 要素を指定します。

```
// where #preview is an existing <canvas> DOM element on your page
const previewEl = document.getElementById('preview');
client.attachPreview(previewEl);
```

利用可能なデバイスのリストを表示する

キャプチャできるデバイスを確認するには、ブラウザの [MediaDevices.enumerateDevices \(\)](#) メソッドをクエリします。

```
const devices = await navigator.mediaDevices.enumerateDevices();
window.videoDevices = devices.filter((d) => d.kind === 'videoinput');
window.audioDevices = devices.filter((d) => d.kind === 'audioinput');
```

MediaStream デバイスから を取得する

使用可能なデバイスのリストを獲得すると、任意の数のデバイスからストリームを取得できます。例えば、カメラからストリームを取得する `getUserMedia()` メソッドを利用できます。

ストリームをキャプチャするデバイスを指定する場合は、メディア制約の `audio` または `video` セクションで `deviceId` を明示的に設定できます。または、`deviceId` を省略して、ブラウザのプロンプトからユーザーにデバイスを選択させることもできます。

`width` および `height` の制約を使用して、理想的なカメラの解像度を指定することもできます。(これらの制約については、[こちら](#)をご覧ください。) SDK では、ブロードキャストの最大解像度に対応する幅および高さの制約が自動的に適用されます。しかし、ソースを SDK に追加した後ソースのアスペクト比が変更されないよう、これらもお客様ご自身で適用することをお勧めします。

```
const streamConfig = IVSBroadcastClient.BASIC_LANDSCAPE;
...
window.cameraStream = await navigator.mediaDevices.getUserMedia({
  video: {
    deviceId: window.videoDevices[0].deviceId,
    width: {
      ideal: streamConfig.maxResolution.width,
    },
    height: {
      ideal: streamConfig.maxResolution.height,
    },
  },
});
window.microphoneStream = await navigator.mediaDevices.getUserMedia({
  audio: { deviceId: window.audioDevices[0].deviceId },
});
```

ストリームにデバイスを追加する

ストリームを取得したら、一意の名前 (以下、camera1) とコンポジションの位置 (ビデオの場合) を指定して、レイアウトにデバイスを追加できます。例えば、ウェブカメラのデバイスを指定すると、そのビデオソースがブロードキャストストリームに追加されます。

ビデオ入力デバイスを指定する際は、配信する「レイヤー」を表すインデックスを指定する必要があります。これは画像編集や CSS と同義で、z-index はレンダリングするレイヤーの順序を表します。オプションで、ストリームソースの X/Y 座標 (サイズと同様) を定義する位置を指定できます。

パラメータの詳細については、「」を参照してください[VideoComposition](#)。

```
client.addVideoInputDevice(window.cameraStream, 'camera1', { index: 0 }); // only
  'index' is required for the position parameter
client.addAudioInputDevice(window.microphoneStream, 'mic1');
```

ブロードキャストを開始する

ブロードキャストを開始するには、Amazon IVS チャンルのストリームキーを指定します。

```
client
  .startBroadcast(streamKey)
  .then((result) => {
    console.log('I am successfully broadcasting!');
  })
  .catch((error) => {
    console.error('Something drastically failed while broadcasting!', error);
  });
```

ブロードキャストを停止する

```
client.stopBroadcast();
```

ビデオの位置をスワップする

クライアントを使用して、ビデオデバイスのコンポジションの位置をスワップできます。

```
client.exchangeVideoDevicePositions('camera1', 'camera2');
```

オーディオをミュート

オーディオをミュートするには、`removeAudioInputDevice` を使用してオーディオデバイスを削除するか、オーディオトラックの `enabled` プロパティを設定します。

```
let audioStream = client.getAudioInputDevice(AUDIO_DEVICE_NAME);
audioStream.getAudioTracks()[0].enabled = false;
```

`AUDIO_DEVICE_NAME` は、`addAudioInputDevice()` の呼び出し中に元のオーディオデバイスに付けられた名前です。

ミュートを解除するには:

```
let audioStream = client.getAudioInputDevice(AUDIO_DEVICE_NAME);
audioStream.getAudioTracks()[0].enabled = true;
```

動画を非表示にする

動画を非表示にするには、`removeVideoInputDevice` を使用してビデオデバイスを削除するか、ビデオトラックの `enabled` プロパティを設定します。

```
let videoStream = client.getVideoInputDevice(VIDEO_DEVICE_NAME).source;
videoStream.getVideoTracks()[0].enabled = false;
```

`VIDEO_DEVICE_NAME` は、元の `addVideoInputDevice()` の呼び出し中にビデオデバイスに付けられた名前です。

再表示するには:

```
let videoStream = client.getVideoInputDevice(VIDEO_DEVICE_NAME).source;
videoStream.getVideoTracks()[0].enabled = true;
```

既知の問題と回避策

- Web Broadcast SDK では、ポート 4443 を開く必要があります。VPN とファイアウォールによってポート 4443 がブロックされ、ストリーミングが妨げられることがあります。

回避策: VPN を無効にするか、ポート 4443 がブロックされないようにファイアウォールを設定します。

- ランドスケープモードからポートレートモードに切り替える場合、バグが多く発生します。

回避策: 該当なし。

- HLS マニフェストで報告されている解像度が正しくありません。この解像度は通常、最初に受信されるものとして、想定よりもはるかに低く設定されています。これには、webRTC の接続中に発生するアップスケーリングが反映されていません。

回避策: 該当なし。

- 最初のページがロードされた後に作成されるクライアントインスタンスが、最初とは異なる `maxFramerate` 設定に応答しない場合があります。

回避方法: 最初のクライアントインスタンスが作成されたときに、`IVSBroadcastClient.create` 関数を使用して `StreamConfig` を一度だけ設定します。

- iOS では、複数のビデオデバイスソースのキャプチャはではサポートされていません WebKit。

回避方法: [この問題](#)に従い、開発プロセスを追跡してください。

- iOS では、ビデオソースが既にある場合に `getUserMedia()` を呼び出すと、`getUserMedia()` を使用して取得した他のすべてのビデオソースが停止します。

回避策: 該当なし。

- WebRTC では、利用可能なリソースに最適なビットレートと解像度が動的に選択されます。ハードウェアまたはネットワークがそれに対応していない場合、ストリームの品質は高くありません。配信中に利用可能なリソースが増えたり少なくなると、ストリームの品質が変化することがあります。

回避方法: 少なくとも 200 kbps のアップロードを行います。

- チャンネルで Amazon S3 への自動録画が有効になっており、Web Broadcast SDK が使用されている場合、同じ S3 プレフィックスへの録画が機能しない可能性があります。これは、Web Broadcast SDK がビットレートと品質を動的に変更するためです。

回避策: 該当なし。

- Next.js を使用する場合、SDK のインポート方法によっては `Uncaught ReferenceError: self is not defined` エラーが発生する場合があります。

回避策: Next.js を使用するときは、[ライブラリを動的にインポートします](#)。

- `module` というタイプ (例: `<script type="module" src="...">`) のスクリプトタグを使用してモジュールをインポートできない場合があります。

回避策: ライブラリには ES6 ビルドがありません。スクリプトタグから `type="module"` を削除します。

- Safari ブロードキャストの視聴者に、ビデオフィールドに緑色のアーティファクトが表示されることがあります。

回避策: ユーザーメディアを再初期化するか、ブロードキャストを再開してください。

Safari での制限事項

- アクセス許可のプロンプトを拒否するには、Safari でのウェブサイト設定のアクセス許可を OS レベルでリセットする必要があります。
- Safari は、Firefox や Chrome ほど効果的にすべてのデバイスをネイティブに検出されません。例えば、OBS 仮想カメラは検知されません。

Firefox での制限事項

- Firefox で画面を共有するには、システムのアクセス許可を有効にする必要があります。有効にした後、正常に動作するには Firefox を再起動する必要があります。再起動しないと、アクセス許可がブロックされていると認識されると、ブラウザは `NotFoundError` 例外をスローします。
- `getCapabilities` メソッドがありません。これは、ユーザーがメディアトラックの解像度やアスペクト比を取得できないことを意味します。こちらの [Bugzilla のスレッド](#) を参照してください。
- レイテンシーやチャンネル数などの、いくつかの `AudioContext` プロパティがありません。これは、オーディオトラックを操作する上級ユーザーにとって問題になる可能性があります。

- getUserMedia からのカメラフィードは、MacOS でのアスペクト比が 4:3 に制限されています。 [Bugzilla のスレッド 1](#) および [Bugzilla のスレッド 2](#) を参照してください。
- オーディオキャプチャが getDisplayMedia でサポートされていません。こちらの [Bugzilla のスレッド](#) を参照してください。
- スクリーンキャプチャのフレームレートが最適ではありません (約 15 fps?)。こちらの [Bugzilla のスレッド](#) を参照してください。

IVS Broadcast SDK: Android ガイド (低レイテンシーストリーミング)

IVS 低レイテンシーストリーミングの Android Broadcast SDK は、Android 上で IVS にブロードキャストするために必要なインターフェイスを提供します。

com.amazonaws.ivs.broadcast パッケージは、このドキュメントで説明されているインターフェイスを実装します。以下のオペレーションがサポートされています。

- ブロードキャストセッションをセットアップ (初期化) します。
- ブロードキャストの管理します。
- 入力デバイスのアタッチとデタッチします。
- コンポジションセッションを管理します。
- イベントを受信します。
- エラーを受信します。

Android Broadcast SDK の最新バージョン : 1.14.1 ([リリースノート](#))

リファレンスドキュメント : Amazon IVS Android Broadcast SDK で使用できる最も重要な方法については、<https://aws.github.io/amazon-ivs-broadcast-docs/1.14.1/android/> のリファレンスドキュメントを参照してください。

サンプルコード: : GitHub <https://github.com/aws-samples/amazon-ivs-broadcast-android-sample> の Android サンプルリポジトリを参照してください。

プラットフォームの要件: Android 5.0 (Lollipop) 以上

開始方法

ライブラリのインストール

Amazon IVS Android Broadcast ライブラリを Android 開発環境に追加するには、ライブラリを以下に示すモジュールの `build.gradle` ファイル (最新バージョンの Amazon IVS Broadcast SDK) に追加します。

```
repositories {
    mavenCentral()
}

dependencies {
    implementation 'com.amazonaws:ivs-broadcast:1.14.1'
}
```

または、SDK を手動でインストールするには、次の場所から最新バージョンをダウンロードします。

<https://search.maven.org/artifact/com.amazonaws/ivs-broadcast>

イベントリスナーの作成

イベントリスナーを設定すると、状態更新、デバイス変更通知、エラー、およびセッションオーディオ情報を受信できます。

```
BroadcastSession.Listener broadcastListener =
    new BroadcastSession.Listener() {
        @Override
        public void onStateChanged(@NonNull BroadcastSession.State state) {
            Log.d(TAG, "State=" + state);
        }

        @Override
        public void onError(@NonNull BroadcastException exception) {
            Log.e(TAG, "Exception: " + exception);
        }
    };
```

必要なアクセス許可

アプリはユーザーのカメラとマイクへのアクセス許可を要求する必要があります。(これは、Amazon IVS に特有なものではなく、カメラやマイクにアクセスする必要があるアプリケーションには必須です。)

ここでは、ユーザーがすでにアクセス許可を付与しているかどうかを確認し、付与していない場合は、許可を求めます。

```
final String[] requiredPermissions =
    { Manifest.permission.CAMERA, Manifest.permission.RECORD_AUDIO };

for (String permission : requiredPermissions) {
    if (ContextCompat.checkSelfPermission(this, permission)
        != PackageManager.PERMISSION_GRANTED) {
        // If any permissions are missing we want to just request them all.
        ActivityCompat.requestPermissions(this, requiredPermissions, 0x100);
        break;
    }
}
```

ここでは、ユーザーの応答を取得します。

```
@Override
public void onRequestPermissionsResult(int requestCode,
    @NonNull String[] permissions,
    @NonNull int[] grantResults) {
    super.onRequestPermissionsResult(requestCode,
        permissions, grantResults);
    if (requestCode == 0x100) {
        for (int result : grantResults) {
            if (result == PackageManager.PERMISSION_DENIED) {
                return;
            }
        }
        setupBroadcastSession();
    }
}
```

ブロードキャストセッションの作成

ブロードキャストインターフェースは `com.amazonaws.ivs.broadcast.BroadcastSession` です。以下に示すように、プリセットを使用して初期化します。初期化中にエラーが発生した場合 (コーデックの設定に失敗など)、`BroadcastListener` にエラーメッセージが表示され、`broadcastSession.isReady` が `false` になります。

重要: Amazon IVS Broadcast SDK for Android へのすべての呼び出しは、SDK がインスタンス化されたスレッドで行う必要があります。別のスレッドからの呼び出しにより、SDK は致命的なエラーをスローし、ブロードキャストを停止します。

```
// Create a broadcast-session instance and sign up to receive broadcast
// events and errors.
Context ctx = getApplicationContext();
broadcastSession = new BroadcastSession(ctx,
    broadcastListener,
    Presets.Configuration.STANDARD_PORTRAIT,
    Presets.Devices.FRONT_CAMERA(ctx));
```

また、「[ブロードキャストセッションの作成 \(詳細バージョン\)](#)」も参照してください。

プレビュー用に ImagePreviewView を設定する

アクティブなカメラデバイスのプレビューを表示する場合は、デバイスのプレビュー `ImagePreviewView` をビュー階層に追加します。

```
// awaitDeviceChanges will fire on the main thread after all pending devices
// attachments have been completed
broadcastSession.awaitDeviceChanges(() -> {
    for(Device device: session.listAttachedDevices()) {
        // Find the camera we attached earlier
        if(device.getDescriptor().type == Device.Descriptor.DeviceType.CAMERA) {
            LinearLayout previewHolder = findViewById(R.id.previewHolder);
            ImagePreviewView preview = ((ImageDevice)device).getPreviewView();
            preview.setLayoutParams(new LinearLayout.LayoutParams(
                LinearLayout.LayoutParams.MATCH_PARENT,
                LinearLayout.LayoutParams.MATCH_PARENT));
            previewHolder.addView(preview);
        }
    }
});
```

ブロードキャストを開始する

GetChannel エンドポイントの ingestEndpoint レスポンスフィールドで受信するホスト名には、rtmps:// を先頭に追加し、/app を末尾に追加する必要があります。URL は、rtmps://{ ingestEndpoint }/app の形式にする必要があります。

```
broadcastSession.start(IVS_RTMP_URL, IVS_STREAMKEY);
```

Android Broadcast SDK は RTMPS 取り込みのみをサポートしています (安全ではない RTMP 取り込みはサポートしていません)。

ブロードキャストを停止する

```
broadcastSession.stop();
```

ブロードキャストセッションを解放する

broadcastSession.release() メソッドは、ブロードキャストセッションが使用されなくなったときに呼び出され、ライブラリが使用するリソースを解放する必要があります。

```
@Override
protected void onDestroy() {
    super.onDestroy();
    previewHolder.removeAllViews();
    broadcastSession.release();
}
```

高度なユースケース

ここでは、いくつかの高度なユースケースを紹介します。上記の基本的な設定から始まり、ここまで進んできました。

ブロードキャスト設定の作成

2つのミキサースロットを持つカスタム設定を作成し、2つの動画ソースをミキサーにバインドできます。1つの画面 (custom) は全画面表示で、右下にある別の小画面 (camera) の後ろに配置されます。custom スロットには、位置、サイズ、またはアスペクトモードを設定しないことに注意してください。これらのパラメータは設定しないので、スロットはサイズと位置の動画設定を使用します。

```
BroadcastConfiguration config = BroadcastConfiguration.with($ -> {
    $.audio.setBitrate(128_000);
    $.video.setMaxBitrate(3_500_000);
    $.video.setMinBitrate(500_000);
    $.video.setInitialBitrate(1_500_000);
    $.video.setSize(1280, 720);
    $.mixer.slots = new BroadcastConfiguration.Mixer.Slot[] {
        BroadcastConfiguration.Mixer.Slot.with(slot -> {
            // Do not automatically bind to a source
            slot.setPreferredAudioInput(
                Device.Descriptor.DeviceType.UNKNOWN);
            // Bind to user image if unbound
            slot.setPreferredVideoInput(
                Device.Descriptor.DeviceType.USER_IMAGE);
            slot.setName("custom");
            return slot;
        }),
        BroadcastConfiguration.Mixer.Slot.with(slot -> {
            slot.setIndex(1);
            slot.setAspect(BroadcastConfiguration.AspectMode.FILL);
            slot.setSize(300, 300);
            slot.setPosition($.video.getSize().x - 350,
                $.video.getSize().y - 350);
            slot.setName("camera");
            return slot;
        })
    };
    return $;
});
```

ブロードキャストセッションの作成 (詳細バージョン)

[基本の例](#)で行ったように `BroadcastSession` を作成しますが、ここでカスタム設定を提供します。デバイスアレイにも `null` を提供します。それらを手動で追加します。

```
// Create a broadcast-session instance and sign up to receive broadcast
// events and errors.
Context ctx = getApplicationContext();
broadcastSession = new BroadcastSession(ctx,
    broadcastListener,
    config, // The configuration we created above
    null); // We'll manually attach devices after
```

カメラデバイスの繰り返し処理とアタッチ

SDK が検出した入力デバイスを繰り返し処理します。Android 7 (Nougat) では、これはデフォルトのマイクデバイスのみを返します。これは、Amazon IVS Broadcast SDK はこのバージョンの Android でデフォルト以外のデバイスの選択をサポートしていないためです。

使用したいデバイスを見つけたら、`attachDevice` を呼び出してアタッチします。入力デバイスの接続が完了すると、メインスレッドで Lambda 関数が呼び出されます。エラーが発生した場合、Listener でエラーが発生します。

```
for(Device.Descriptor desc:
BroadcastSession.listAvailableDevices(getApplicationContext())) {
    if(desc.type == Device.Descriptor.DeviceType.CAMERA &&
        desc.position == Device.Descriptor.Position.FRONT) {
        session.attachDevice(desc, device -> {
            LinearLayout previewHolder = findViewById(R.id.previewHolder);
            ImagePreviewView preview = ((ImageDevice)device).getPreviewView();
            preview.setLayoutParams(new LinearLayout.LayoutParams(
                LinearLayout.LayoutParams.MATCH_PARENT,
                LinearLayout.LayoutParams.MATCH_PARENT));
            previewHolder.addView(preview);
            // Bind the camera to the mixer slot we created above.
            session.getMixer().bind(device, "camera");
        });
        break;
    }
}
```

カメラのスワップ

```
// This assumes you've kept a reference called "currentCamera" that points to
// a front facing camera
for(Device device: BroadcastSession.listAvailableDevices()) {
    if(device.type == Device.Descriptor.DeviceType.CAMERA &&
        Device.position != currentCamera.position) {
        // Remove the preview view for the old device.
        // setImagePreviewTextureView is an example function
        // that handles your view hierarchy.
        setImagePreviewView(null);
        session.exchangeDevices(currentCamera, device, camera -> {
            // Set the preview view for the new device.
            setImagePreviewView(camera.getPreviewView());
        });
    }
}
```

```
        currentCamera = camera;
    });
    break;
}
}
```

入力サーフェスを作成する

アプリが生成するサウンドや画像データを入力するには、`createImageInputSource` または `createAudioInputSource` を使用します。どちらの方法も、他のデバイスと同様にミキサにバインドできる仮想デバイスを作成してアタッチします。

`createImageInputSource` が返す `SurfaceSource` には `getInputSurface` メソッドが含まれます。このメソッドは、Camera2 API、OpenGL、Vulkan、または Surface に書き込みできるその他のもので使用できる Surface を提供します。

によって `AudioDevice` 返される `createAudioInputSource` は、`AudioRecorder` またはその他の方法で生成された線形 PCM データを受信できます。

```
SurfaceSource source = session.createImageInputSource();
Surface surface = source.getInputSurface();
session.getMixer().bind(source, "custom");
```

デバイスのデタッチ

デバイスをデタッチし、交換しない場合は、`Device` または `Device.Descriptor` を使用してデタッチします。

```
session.detachDevice(currentCamera);
```

画面およびシステムオーディオキャプチャ

Amazon IVS Broadcast SDK for Android には、デバイスの画面 (Android 5 以降) とシステムオーディオ (Android 10 以降) のキャプチャの簡素化に役立ちます。これらを手動で管理する場合は、カスタム画像入力ソースとカスタムオーディオ入力ソースを作成できます。

画面およびシステムオーディオのキャプチャセッションを作成するには、最初にアクセス許可リクエストのインテントを作成する必要があります。

```
public void startScreenCapture() {
```

```
MediaProjectionManager manager =
    (MediaProjectionManager) getApplicationContext()
        .getSystemService(Context.MEDIA_PROJECTION_SERVICE);
if(manager != null) {
    Intent intent = manager.createScreenCaptureIntent();
    startActivityIfNeeded(intent, SCREEN_CAPTURE_REQUEST_ID);
}
}
```

この機能を使用するには、`com.amazonaws.ivs.broadcast.SystemCaptureService` を拡張するクラスを提供する必要があります。それらのメソッドをオーバーライドする必要はありませんが、サービス間の潜在的な衝突を避けるためにクラスが必要になります。

また、Android マニフェストにいくつかの要素を追加する必要があります。

```
<uses-permission android:name="android.permission.FOREGROUND_SERVICE" />
<application ...>
    <service android:name=".ExampleSystemCaptureService"
        android:foregroundServiceType="mediaProjection"
        android:isolatedProcess="false" />
</application>
...
```

`SystemCaptureService` を拡張するクラスは、`<service>` 要素で名前を付ける必要があります。Android 9 以降では、`foregroundServiceType` は `mediaProjection` でなければなりません。

アクセス許可のインテントを受け取ると、画面とシステムオーディオのキャプチャセッションの作成に進めます。Android 8 以降では、ユーザーの通知パネルに表示される通知を提供する必要があります。Amazon IVS Broadcast SDK for Android は、便利なメソッド `createServiceNotificationBuilder` を提供します。代わりに、独自の通知を提供することもできます。

```
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    if(requestCode != SCREEN_CAPTURE_REQUEST_ID
        || Activity.RESULT_OK != resultCode) {
        return;
    }
    Notification notification = null;
```

```
if(Build.VERSION.SDK_INT >= 26) {
    Intent intent = new Intent(getApplicationContext(),
                                NotificationActivity.class);
    notification = session
        .createServiceNotificationBuilder("example",
                                         "example channel", intent)
        .build();
}
session.createSystemCaptureSources(data,
    ExampleSystemCaptureService.class,
    Notification,
    devices -> {
        // This step is optional if the mixer slots have been given preferred
        // input device types SCREEN and SYSTEM_AUDIO
        for (Device device : devices) {
            session.getMixer().bind(device, "game");
        }
    });
}
```

推奨されるブロードキャスト設定の取得

ブロードキャストを開始する前にユーザーの接続を評価するには、`recommendedVideoSettings` メソッドを使用して簡単なテストを実行します。テストが実行されると、最も推奨されるものから順に、さまざまな推奨事項が表示されます。このバージョンの SDK では、現在の `BroadcastSession` を再構成することはできないため、`release()` してから、推奨設定で新しいものを作成する必要があります。 `Result.status` が `SUCCESS` または `ERROR` になるまで、`BroadcastSessionTest.Results` を受け取り続けます。進捗状況を確認するには `Result.progress` を使用します。

Amazon IVS は、最大ビットレート 8.5 MBps (type が `STANDARD` または `ADVANCED` のチャンネルの場合) をサポートしているため、このメソッドが返す `maximumBitrate` が 8.5 MBps を超えることは一切ありません。ネットワークパフォーマンスのわずかな変動を考慮するために、このメソッドが返す奨励される `initialBitrate` はテストで測定した実際のビットレートよりわずかに小さくなります。(通常、使用可能な帯域幅の 100% を使用することはお勧めできません。)

```
void runBroadcastTest() {
    this.test = session.recommendedVideoSettings(RTMP_ENDPOINT, RTMP_STREAMKEY,
        result -> {
            if (result.status == BroadcastSessionTest.Status.SUCCESS) {
                this.recommendation = result.recommendations[0];
            }
        });
}
```

```
    }  
    });  
}
```

Bluetooth マイクの使用

Bluetooth マイクデバイスを使用してブロードキャストするには、Bluetooth SCO 接続を開始する必要があります。

```
Bluetooth.startBluetoothSco(context);  
// Now bluetooth microphones can be used  
...  
// Must also stop bluetooth SCO  
Bluetooth.stopBluetoothSco(context);
```

既知の問題と回避策

- Bluetooth 経由で接続された外部マイクを使用すると、不安定になる可能性があります。ブロードキャストセッション中に Bluetooth デバイスを接続または切断すると、デバイスが確実にデタッチされて再アタッチされるまで、マイク入力が機能しなくなることがあります。

回避策: Bluetooth ヘッドセットを使用する場合は、ブロードキャストを開始する前に接続し、ブロードキャスト全体を通して接続したままにします。

- Broadcast SDK は、USB 経由で接続された外部カメラへのアクセスをサポートしません。

回避方法: USB 経由で接続された外部カメラは使用しないでください。

- (カスタムオーディオソースを使用して) 実際の速度よりも高速にオーディオデータを送信すると、オーディオドリフトが発生します。

回避方法: 実際の速度よりも高速にオーディオデータを送信しないでください。

- 同じ BroadcastSession が複数のブロードキャストに使用されている場合、一部の Android 5 デバイスは黒い画像をストリーミングすることがあります。

回避方法: BroadcastSession を停止したときに、それを解放し、新しいものを初期化します。

- Android 5、6、および 7 デバイスで許可されるのがシステムのデフォルトのマイクのみであるため、これらの Android バージョンではマイクに対する Broadcast SDK の onDeviceAdded コールバックと onDeviceRemoved コールバックを受信できません。

回避策: これらのデバイスでは、Broadcast SDK はシステムのデフォルトのマイクを使用します。

- ImagePreviewView が親から削除された場合 (例えば、removeView() が親側で呼び出された場合)、ImagePreviewView はすぐにリリースされます。ImagePreviewView を別の親ビューに追加しても、フレームは表示されません。

回避策: getPreview を使用して別のプレビューをリクエストします。

- 一部の Android ビデオエンコーダーは、ビデオサイズが 176 x 176 未満の場合は設定できません。サイズを小さく設定するとエラーが発生し、ストリーミングが妨げられます。

回避策: ビデオサイズを 176 x 176 未満に設定しないでください。

IVS Broadcast SDK: iOS ガイド (低レイテンシーストリーミング)

IVS 低レイテンシーストリーミング iOS Broadcast SDK は、iOS 上の Amazon IVS へのブロードキャストに必要なインターフェイスを提供します。

AmazonIVSBroadcast モジュールは、このドキュメントで説明されているインターフェイスを実装します。以下のオペレーションがサポートされています。

- ブロードキャストセッションをセットアップ (初期化) します。
- ブロードキャストの管理します。
- 入力デバイスのアタッチとデタッチします。
- コンポジションセッションを管理します。
- イベントを受信します。
- エラーを受信します。

iOS Broadcast SDK の最新バージョン: 1.14.1 ([リリースノート](#))

リファレンスドキュメント: Amazon IVS iOS Broadcast SDK で使用できる最も重要な方法については、<https://aws.github.io/amazon-ivs-broadcast-docs/1.14.1/ios/> のリファレンスドキュメントを参照してください。

サンプルコード: : GitHub<https://github.com/aws-samples/amazon-ivs-broadcast-ios-sample> の iOS サンプルリポジトリを参照してください。

プラットフォームの要件: iOS 12 以上

開始方法

ライブラリのインストール

Broadcast SDK は 経由で統合することをお勧めします CocoaPods。 (代わりに、フレームワークを手動でプロジェクトに追加することも可能です)。

推奨: Broadcast SDK の統合 (CocoaPods)

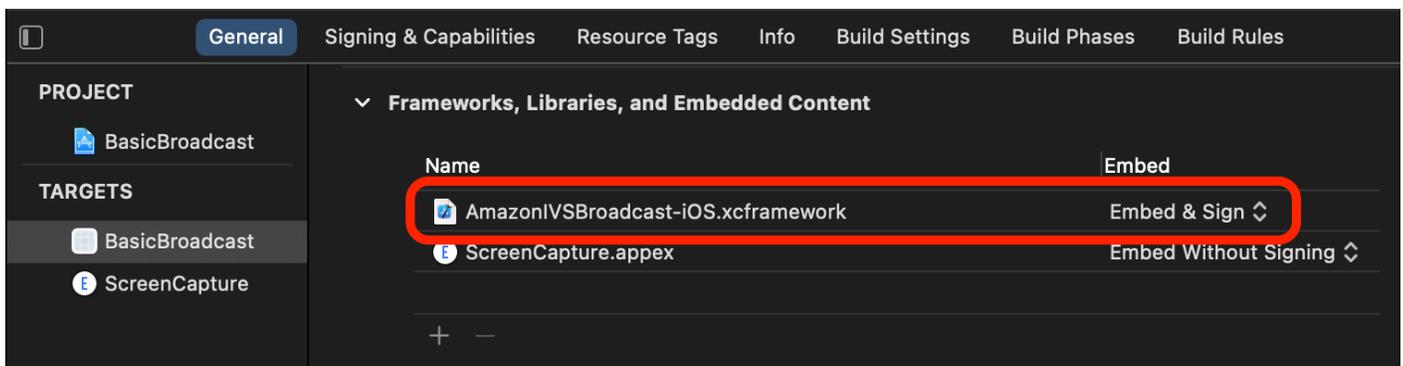
リリースは、 という名前 CocoaPods の で公開されます AmazonIVSBroadcast。 この依存関係を自分の Podfile に追加します。

```
pod 'AmazonIVSBroadcast'
```

pod install を実行すると、SDK が .xcworkspace で利用できるようになります。

代替方法: フレームワークを手動でインストールする

1. <https://broadcast.live-video.net/1.14.1/AmazonIVSBroadcast.xcframework.zip> から最新バージョンをダウンロードします。
2. アーカイブの内容を抽出します。 AmazonIVSBroadcast.xcframework には、デバイスとシミュレータの両方の SDK が含まれています。
3. アプリケーションターゲットの [全般] タブの、 [Frameworks, Libraries, and Embedded Content (フレームワーク、ライブラリ、埋め込みコンテンツ)] のセクションに AmazonIVSBroadcast.xcframework をドラッグして埋め込みます。



IVS の実装 BroadcastSession.Delegate

IVSBroadcastSession.Delegate を実装します。これにより、状態の更新やデバイス変更の通知を受け取ることができます。

```
extension ViewController : IVSBroadcastSession.Delegate {
    func broadcastSession(_ session: IVSBroadcastSession,
                          didSetChange state: IVSBroadcastSession.State) {
        print("IVSBroadcastSession did change state \(state)")
    }

    func broadcastSession(_ session: IVSBroadcastSession,
                          didSetEmitError error: Error) {
        print("IVSBroadcastSession did emit error \(error)")
    }
}
```

必要なアクセス許可

アプリはユーザーのカメラとマイクへのアクセス許可を要求する必要があります。(これは、Amazon IVS に特有なものではなく、カメラやマイクにアクセスする必要があるアプリケーションには必須です。)

ここでは、ユーザーがすでにアクセス許可を付与しているかどうかを確認し、付与していない場合は、許可を求めます。

```
switch AVCaptureDevice.authorizationStatus(for: .video) {
case .authorized: // permission already granted.
case .notDetermined:
    AVCaptureDevice.requestAccess(for: .video) { granted in
        // permission granted based on granted bool.
    }
case .denied, .restricted: // permission denied.
@unknown default: // permissions unknown.
}
```

カメラやマイクにアクセスするには、`.video` と `.audio` の両方のメディアタイプに対してこれを行う必要があります。

また、`NSCameraUsageDescription` と `NSMicrophoneUsageDescription` のエントリを `Info.plist` に追加する必要があります。これを行わずにアクセス許可をリクエストすると、アプリがクラッシュします。

アプリケーションアイドルタイマーの無効化

これはオプションですが推奨されます。Broadcast SDK の使用中にデバイスがスリープ状態になり、ブロードキャストが中断されるのを防ぎます。

```
override fun viewDidAppear(_ animated: Bool) {
    super.viewDidAppear(animated)
    UIApplication.shared.isIdleTimerDisabled = true
}
override fun viewWillDisappear(_ animated: Bool) {
    super.viewWillDisappear(animated)
    UIApplication.shared.isIdleTimerDisabled = false
}
```

(オプション) AV の設定AudioSession

デフォルトでは、Broadcast SDK がアプリケーション AVAudioSession をセットアップします。これを自分で管理する場合は、IVSBroadcastSession.applicationAudioSessionStrategy を noAction に設定します。AVAudioSession の制御がなければ、Broadcast SDK はマイクロホンを実験的に管理できません。オプション noAction でマイクを使用するには、IVSCustomAudioSource を作成し、AVCaptureSession、AVAudioEngine、または PCM オーディオサンプルを提供する別のツールを介して独自のサンプルを提供できます。

手動で AVAudioSession をセットアップするには、少なくとも、カテゴリを .record または .playbackAndRecord に設定し、それを active に設定する必要があります。Bluetooth デバイスからオーディオを録音する場合は、.allowBluetooth オプションも設定する必要があります。

```
do {
    try AVAudioSession.sharedInstance().setCategory(.record, options: .allowBluetooth)
    try AVAudioSession.sharedInstance().setActive(true)
} catch {
    print("Error configuring AVAudioSession")
}
```

SDK にこれを処理させることをお勧めします。異なるオーディオデバイスを選択する場合は、ポートを手動で管理する必要があります。

ブロードキャストセッションの作成

ブロードキャストインターフェースは IVSBroadcastSession です。以下に示すように、インターフェースを初期化します。

```
let broadcastSession = try IVSBroadcastSession(
    configuration: IVSPresets.configurations().standardLandscape(),
    descriptors: IVSPresets.devices().frontCamera(),
    delegate: self)
```

「[ブロードキャストセッションの作成 \(詳細バージョン\)](#)」も参照してください。

プレビューImagePreviewView 用に IVS を設定する

アクティブなカメラデバイスのプレビューを表示する場合は、デバイスのプレビュー IVSImagePreviewView をビュー階層に追加します。

```
// If the session was just created, execute the following
// code in the callback of IVSBroadcastSession.awaitDeviceChanges
// to ensure all devices have been attached.
if let devicePreview = try broadcastSession.listAttachedDevices()
    .compactMap({ $0 as? IVSImageDevice })
    .first?
    .previewView()
{
    previewView.addSubview(devicePreview)
}
```

ブロードキャストを開始する

GetChannel エンドポイントの ingestEndpoint レスポンスフィールドで受信するホスト名には、rtmps:// を先頭に追加し、/app を末尾に追加する必要があります。URL は、rtmps://{ ingestEndpoint }/app の形式にする必要があります。

```
try broadcastSession.start(with: IVS_RTMP_URL, streamKey: IVS_STREAMKEY)
```

iOS Broadcast SDK は RTMPS 取り込みのみをサポートしています (安全ではない RTMP 取り込みはサポートしていません)。

ブロードキャストを停止する

```
broadcastSession.stop()
```

ライフサイクルイベントの管理

オーディオの中断

Broadcast SDK がオーディオ入力ハードウェアに排他的にアクセスできないシナリオがいくつかあります。処理する必要があるシナリオの例は、次のとおりです。

- ユーザーが電話または FaceTime 通話を受信する
- ユーザーが Siri をアクティベートする

Apple では、`AVAudioSession.interruptionNotification` に登録することで、これらのイベントに簡単に対応できるようにしています。

```
NotificationCenter.default.addObserver(  
    self,  
    selector: #selector(audioSessionInterrupted(_:)),  
    name: AVAudioSession.interruptionNotification,  
    object: nil)
```

次のような方法でイベントを処理できます。

```
// This assumes you have a variable `isRunning` which tracks if the broadcast is  
// currently live, and another variable `wasRunningBeforeInterruption` which tracks  
// whether the broadcast was active before this interruption to determine if it should  
// resume after the interruption has ended.  
  
@objc  
private func audioSessionInterrupted(_ notification: Notification) {  
    guard let userInfo = notification.userInfo,  
          let typeValue = userInfo[AVAudioSessionInterruptionTypeKey] as? UInt,  
          let type = AVAudioSession.InterruptionType(rawValue: typeValue)  
    else {  
        return  
    }  
    switch type {  
    case .began:  
        wasRunningBeforeInterruption = isRunning  
        if isRunning {  
            broadcastSession.stop()  
        }  
    case .ended:
```

```
defer {
    wasRunningBeforeInterruption = false
}
guard let optionsValue = userInfo[AVAudioSessionInterruptionOptionKey] as? UInt
else { return }
let options = AVAudioSession.InterruptionOptions(rawValue: optionsValue)
if options.contains(.shouldResume) && wasRunningBeforeInterruption {
    try broadcastSession.start(
        with: IVS_RTMP_URL,
        streamKey: IVS_STREAMKEY)
}
@unknown default: break
}
}
```

バックグラウンドに移行したアプリケーションについて

iOS の標準アプリケーションは、バックグラウンドでカメラを使用することはできません。バックグラウンドでの動画エンコーディングにも制限があります。ハードウェアエンコーダには制限があるため、フォアグラウンドアプリケーションだけがアクセス可能です。このため、Broadcast SDK は自動的にセッションを終了し、isReady プロパティを false に設定します。アプリケーションがフォアグラウンドに再び移行しようとする時、Broadcast SDK がすべてのデバイスを元の IVSMixerSlotConfiguration エントリに再アタッチします。

Broadcast SDK は、UIApplication.didEnterBackgroundNotification と UIApplication.willEnterForegroundNotification に応答することでこれを行います。

カスタムイメージソースを提供する場合は、これらの通知を処理する準備が必要です。ストリームが終了する前に、それらを破棄するために追加の手順の実行が必要になる場合があります。

アプリケーションがバックグラウンドにあるときにストリーミングを有効にする回避策については、「[バックグラウンドビデオの使用](#)」を参照してください。

メディアサービスの消失

ごくまれに、iOS デバイス上のメディアサブシステム全体がクラッシュすることがあります。このシナリオでは、ブロードキャストすることができなくなります。これらの通知に適切に応答するかどうかは、アプリケーションに依存します。少なくとも、次の通知を登録してください。

- [mediaServicesWereLostNotification](#) — ブロードキャストを停止し、 の割り当てを完全に解除することで応答しますIVSBroadcastSession。ブロードキャストセッションが使用するすべての内部コンポーネントが無効になります。

- [mediaServicesWereResetNotification](#) - ユーザーに再度ブロードキャストできることを通知することとで応答します。ユースケースによっては、この時点でブロードキャストを自動的に開始することができます。

高度なユースケース

ここでは、いくつかの高度なユースケースを紹介します。上記の基本的な設定から始まり、ここまで進んできました。

ブロードキャスト設定の作成

2つのミキサースロットを持つカスタム設定を作成し、2つの動画ソースをミキサーにバインドできます。1つの画面 (custom) は全画面表示で、右下にある別の小画面 (camera) の後ろに配置されます。custom スロットには、位置、サイズ、またはアスペクトモードを設定しないことに注意してください。これらのパラメータは設定しないので、スロットはサイズと位置の動画設定を使用します。

```
let config = IVSBroadcastConfiguration()
try config.audio.setBitrate(128_000)
try config.video.setMaxBitrate(3_500_000)
try config.video.setMinBitrate(500_000)
try config.video.setInitialBitrate(1_500_000)
try config.video.setSize(CGSize(width: 1280, height: 720))
config.video.defaultAspectMode = .fit
config.mixer.slots = [
    try {
        let slot = IVSMixerSlotConfiguration()
        // Do not automatically bind to a source
        slot.preferredAudioInput = .unknown
        // Bind to user image if unbound
        slot.preferredVideoInput = .userImage
        try slot.setName("custom")
        return slot
    }(),
    try {
        let slot = IVSMixerSlotConfiguration()
        slot.zIndex = 1
        slot.aspect = .fill
        slot.size = CGSize(width: 300, height: 300)
        slot.position = CGPoint(x: config.video.size.width - 400, y:
config.video.size.height - 400)
        try slot.setName("camera")
```

```
        return slot
    }()
]
```

ブロードキャストセッションの作成 (詳細バージョン)

[基本的な例](#)で行ったように `IVSBroadcastSession` を作成しますが、ここでカスタム設定を提供します。デバイスアレイにも `nil` を提供します。それらを手動で追加します。

```
let broadcastSession = try IVSBroadcastSession(
    configuration: config, // The configuration we created above
    descriptors: nil, // We'll manually attach devices after
    delegate: self)
```

カメラデバイスの繰り返し処理とアタッチ

SDK が検出した入力デバイスを繰り返し処理します。SDK が iOS の組み込みデバイスだけを返します。Bluetooth オーディオデバイスが接続されていても、内蔵デバイスとして表示されます。詳細については、「[既知の問題と回避策](#)」を参照してください。

使用したいデバイスを見つけたら、アタッチするために `attachDevice` を呼び出します。

```
let frontCamera = IVSBroadcastSession.listAvailableDevices()
    .filter { $0.type == .camera && $0.position == .front }
    .first
if let camera = frontCamera {
    broadcastSession.attach(camera, toSlotWithName: "camera") { device, error in
        // check error
    }
}
```

カメラのスワップ

```
// This assumes you've kept a reference called `currentCamera` that points to the
current camera.
let wants: IVSDevicePosition = (currentCamera.descriptor().position
    == .front) ? .back : .front
// Remove the current preview view since the device will be changing.
previewView.subviews.forEach { $0.removeFromSuperview() }
let foundCamera = IVSBroadcastSession
```

```
.listAvailableDevices()
    .first { $0.type == .camera && $0.position == wants }
guard let newCamera = foundCamera else { return }
broadcastSession.exchangeOldDevice(currentCamera, withNewDevice: newCamera)
{ newDevice, _ in
    currentCamera = newDevice
    if let camera = newDevice as? IVSImageDevice {
        do {
            previewView.addSubview(try finalCamera.previewView())
        } catch {
            print("Error creating preview view \(error)")
        }
    }
}
```

カスタム入力ソースを作成する

アプリが生成するサウンドや画像データを入力するには、`createImageSource` または `createAudioSource` を使用します。どちらの方法も、他のデバイスと同様にミキサーにバインドできる仮想デバイス (`IVSCustomImageSource` および `IVSCustomAudioSource`) を作成します。

これらの両方のメソッドが返すデバイスは、`onSampleBuffer` 関数を介して `CMSampleBuffer` を受け取ります。

- 動画ソースの場合、ピクセルフォーマットは `kCVPixelFormatType_32BGRA`、`420YpCbCr8BiPlanarFullRange`、または `420YpCbCr8BiPlanarVideoRange` です。
- オーディオソースの場合、バッファにはリニア PCM データが含まれている必要があります。

Broadcast SDK が提供するカメラデバイスを使用しているときに、カメラ入力での `AVCaptureSession` を使用してカスタム画像ソースをフィードすることはできません。複数のカメラを同時に使用する場合は、`AVCaptureMultiCamSession` を使用して2つのカスタムイメージソースを提供します。

カスタム画像ソースは、主に画像などの静的コンテンツや動画コンテンツで使用する必要があります。

```
let customImageSource = broadcastSession.createImageSource(withName: "video")
try broadcastSession.attach(customImageSource, toSlotWithName: "custom")
```

ネットワーク接続をモニタリングする

モバイルデバイスは、外出中に一時的にネットワーク接続を失ったり、回復したりするのが普通です。このため、アプリのネットワーク接続をモニタリングし、状況が変化したときに適切に対応することが重要です。

ブロードキャスターの接続が切断されると、Broadcast SDK の状態は `error` に変わり、次に `disconnected` に変わります。これらの状態の変化は、`IVSBroadcastSessionDelegate` を通じて通知されます。これらの状態の変化を受信すると、次のようになります。

1. ブロードキャストアプリの接続状態をモニタリングし、接続が回復すると、エンドポイントとストリームキーを使用して `start` を呼び出します。
2. **重要:** 状態デリゲートコールバックをモニタリングし、`start` を再度呼び出した後、状態が `connected` に変わることを確認します。

デバイスのデタッチ

デバイスをデタッチし、交換しない場合は、`IVSDevice` または `IVSDeviceDescriptor` を使ってデバイスをデタッチします。

```
broadcastSession.detachDevice(currentCamera)
```

ReplayKit 統合

iOS でデバイスの画面とシステムオーディオをストリーミングするには、と統合する必要があります [ReplayKit](#)。Amazon IVS Broadcast SDK を使用すると、`ReplayKit` を使用して簡単に統合できます `IVSReplayKitBroadcastSession`。 `RPBroadcastSampleHandler` サブクラスで `IVSReplayKitBroadcastSession` のインスタンスを作成してから、次を行います。

- `broadcastStarted` でセッションを開始する
- `broadcastFinished` でセッションを停止する

セッションオブジェクトには、画面の画像、アプリオーディオ、マイクオーディオ用の 3 つのカスタムソースがあります。 `processSampleBuffer` で提供されている `CMSampleBuffers` をそれらのカスタムソースに渡します。

デバイスの向きを処理するには、サンプルバッファから `ReplayKit` 固有のメタデータを抽出する必要があります。以下のコードを使用します。

```
let imageSource = session.systemImageSource;
if let orientationAttachment = CMGetAttachment(sampleBuffer, key:
  RPVideoSampleOrientationKey as CFString, attachmentModeOut: nil) as? NSNumber,
  let orientation = CGImagePropertyOrientation(rawValue:
orientationAttachment.uint32Value) {
  switch orientation {
  case .up, .upMirrored:
    imageSource.setHandsetRotation(0)
  case .down, .downMirrored:
    imageSource.setHandsetRotation(Float.pi)
  case .right, .rightMirrored:
    imageSource.setHandsetRotation(-(Float.pi / 2))
  case .left, .leftMirrored:
    imageSource.setHandsetRotation((Float.pi / 2))
  }
}
```

IVSBroadcastSession の代わりに ReplayKit を使用して統合できま
すIVSReplayKitBroadcastSession。ただし、ReplayKit固有のバリエーションには、ブロードキャ
スト拡張のために Apple のメモリ上限内に留まるように、内部メモリフットプリントを削減するた
めのいくつかの変更があります。

推奨されるブロードキャスト設定の取得

ブロードキャストを開始する前にユーザーの接続を評価するに
は、IVSBroadcastSession.recommendedVideoSettings を使用して簡単なテストを実行しま
す。テストが実行されると、最も推奨されるものから順に、さまざまな推奨事項が表示されます。
このバージョンの SDK では、現在の IVSBroadcastSession を再設定することができないため、
割り当てを解除してから、推奨設定で新しい設定を作成する必要があります。result.status が
Success または Error になるまで、IVSBroadcastSessionTestResults を受け取り続けま
す。進捗状況を確認するには result.progress を使用します。

Amazon IVS は、最大ビットレート 8.5 MBps (type が STANDARD または ADVANCED のチャネルの
場合) をサポートしているため、このメソッドが返す maximumBitrate が 8.5 MBps を超えること
は一切ありません。ネットワークパフォーマンスのわずかな変動を考慮するために、このメソッドが
返す奨励される initialBitrate はテストで測定した実際のビットレートよりわずかに小さくなり
ます。(通常、使用可能な帯域幅の 100% を使用することはお勧めできません。)

```
func runBroadcastTest() {
```

```
self.test = session.recommendedVideoSettings(with: IVS_RTMP_URL, streamKey:
IVS_STREAMKEY) { [weak self] result in
    if result.status == .success {
        this.recommendation = result.recommendations[0];
    }
}
}
```

バックグラウンドビデオを使用する

バックグラウンドでアプリケーションを使用しても、非RelayKit ブロードキャストを続行できます。

電力を節約し、フォアグラウンドアプリケーションの応答性を維持するために、iOS では GPU に一度に 1 つのアプリケーションしかアクセスできません。Amazon IVS Broadcast SDK は、複数の入力ソースの合成、イメージのスケーリング、イメージのエンコードなど、ビデオパイプラインの複数のステージで GPU を使用します。ブロードキャストアプリケーションがバックグラウンドにある間は、SDK がこれらのアクションを実行できるという保証はありません。

これに対処するには、`createAppBackgroundImageSource` 方法を使用します。これにより、SDK はバックグラウンドでビデオとオーディオの両方をブロードキャストし続けることができます。IVSBackgroundImageSource を返します。これは、追加の IVSCustomImageSource 関数を持つ通常の `finish` です。バックグラウンドイメージソースに提供されるすべての CMSampleBuffer は、元の IVSVideoConfiguration によって提供されるフレームレートでエンコードされます。CMSampleBuffer のタイムスタンプは無視されます。

SDK は、これらのイメージをスケーリングしてエンコードし、キャッシュし、アプリケーションがバックグラウンドになったときにそのフィードを自動的にループします。アプリケーションがフォアグラウンドに戻ると、アタッチされたイメージデバイスが再びアクティブになり、事前エンコードされたストリームはループを停止します。

このプロセスを元に戻すには、`removeImageSourceOnAppBackgrounded` を使用します。SDK のバックグラウンド動作を明示的に元に戻す場合を除き、これを呼び出す必要はありません。そうしないと、IVSBroadcastSession の割り当て解除時に自動的にクリーンアップされます。

メモ: セッションがライブになる前に、ブロードキャストセッションの設定の一環として、このメソッドを呼び出すことを強くお勧めします。この方法は高価である (ビデオをエンコードする) ため、このメソッドの実行中のライブブロードキャストのパフォーマンスが低下する可能性があります。

例:バックグラウンドビデオのスタティックイメージの生成

バックグラウンドソースに1つのイメージを指定すると、その静的イメージの完全な GOP が生成されます。

CImage を使用する例を次に示します。

```
// Create the background image source
guard let source = session.createAppBackgroundImageSource(withAttemptTrim: true,
  onComplete: { error in
    print("Background Video Generation Done - Error: \(error.debugDescription)")
  }) else {
  return
}

// Create a UIImage of the color red.
let ciImage = UIImage(color: .red)

// Convert the UIImage to a CVPixelBuffer
let attrs = [
  kCVPixelBufferCGImageCompatibilityKey: kCFBooleanTrue,
  kCVPixelBufferCGBitmapContextCompatibilityKey: kCFBooleanTrue,
  kCVPixelBufferMetalCompatibilityKey: kCFBooleanTrue,
] as CFDictionary

var pixelBuffer: CVPixelBuffer!
CVPixelBufferCreate(kCFAllocatorDefault,
  videoConfig.width,
  videoConfig.height,
  kCVPixelFormatType_420YpCbCr8BiPlanarFullRange,
  attrs,
  &pixelBuffer)

let context = CGContext()
context.render(ciImage, to: pixelBuffer)

// Submit to CVPixelBuffer and finish the source
source.add(pixelBuffer)
source.finish()
```

または、単色の CImage を作成する代わりに、バンドルされたイメージを使用することもできます。ここに示す唯一のコードは、前のサンプルで使用するために UIImage を CImage に変換する方法です。

```
// Load the pre-bundled image and get it's CGImage
guard let cgImage = UIImage(named: "image")?.cgImage else {
    return
}

// Create a CIImage from the CGImage
let ciImage = CIImage(cgImage: cgImage)
```

例: AV を使用したビデオAssetImageGenerator

AVAssetImageGenerator を使用して CMSampleBuffers から AVAsset を生成できます (ただし、HLS ストリーム AVAsset ではありません)。

```
// Create the background image source
guard let source = session.createAppBackgroundImageSource(withAttemptTrim: true,
    onComplete: { error in
        print("Background Video Generation Done - Error: \(error.debugDescription)")
    }) else {
    return
}

// Find the URL for the pre-bundled MP4 file
guard let url = Bundle.main.url(forResource: "sample-clip", withExtension: "mp4") else
{
    return
}

// Create an image generator from an asset created from the URL.
let generator = AVAssetImageGenerator(asset: AVAsset(url: url))
// It is important to specify a very small time tolerance.
generator.requestedTimeToleranceAfter = .zero
generator.requestedTimeToleranceBefore = .zero

// At 30 fps, this will generate 4 seconds worth of samples.
let times: [NSNumber] = (0...120).map { NSNumber(time: CMTime(value: $0, timescale:
    CMTimeScale(config.video.targetFramerate))) }
var completed = 0

let context = CIContext(options: [.workingColorSpace: NSNull()])

// Create a pixel buffer pool to efficiently feed the source
let attrs = [
    kCVPixelBufferPixelFormatTypeKey: kCVPixelFormatType_420YpCbCr8BiPlanarFullRange,
    kCVPixelBufferCGImageCompatibilityKey: kCFBooleanTrue,
```

```
kCVPixelBufferCGBitmapContextCompatibilityKey: kCFBooleanTrue,
kCVPixelBufferMetalCompatibilityKey: kCFBooleanTrue,
kCVPixelBufferWidthKey: videoConfig.width,
kCVPixelBufferHeightKey: videoConfig.height,
] as CFDictionary
var pool: CVPixelBufferPool!
CVPixelBufferPoolCreate(kCFAllocatorDefault, nil, attrs, &pool)

generator.generateCGImagesAsynchronously(forTimes: times) { requestTime, image,
actualTime, result, error in
    if let image = image {
        // convert to CIImage then CVPixelBuffer
        let ciImage = CIImage(cgImage: image)
        var pixelBuffer: CVPixelBuffer!
        CVPixelBufferPoolCreatePixelBuffer(kCFAllocatorDefault, pool, &pixelBuffer)
        context.render(ciImage, to: pixelBuffer)
        source.add(pixelBuffer)
    }
    completed += 1
    if completed == times.count {
        // Mark the source finished when all images have been processed
        source.finish()
    }
}
```

CVPixelBuffers および AVPlayer を使用して AVPlayerItemVideoOutput を生成することは可能です。ただし、これには CADisplayLink を使用する必要があり、リアルタイムに近い方法で実行されますが、AVAssetImageGenerator はフレームをはるかに高速に処理できます。

制限事項

バックグラウンドに入った後に中断されないようにするために、アプリケーションには、[バックグラウンドオーディオエンタイトルメント](#)が必要です。

アプリケーションが完了するには GPU にアクセスする必要があるため、createAppBackgroundImageSource は、アプリケーションがフォアグラウンドにあるときにのみ呼び出すことができます。

createAppBackgroundImageSource は常に完全な GOP にエンコードします。たとえば、キーフレーム間隔が 2 秒 (デフォルト) で 30 fps で実行されている場合、60 フレームの倍数をエンコードします。

- 60 フレーム未満を指定すると、トリムオプションの値に関係なく、60 フレームに達するまで最後のフレームが繰り返されます。
- 60 以上のフレームが提供され、トリムオプションが true の場合、最後の N フレームがドロップされます。ここで、N は送信されたフレームの総数の残りを 60 で割った値です。
- 60 以上のフレームが提供され、トリムオプションが false の場合、次の 60 フレームの倍数に達するまで最後のフレームが繰り返されます。

iOS がカメラの解像度とフレームレートを選択する方法

Broadcast SDK によって管理されるカメラは、解像度とフレームレート (frames-per-second、または FPS) を最適化して、電力の生産と消費を最小限に抑えます。このセクションでは、ホストアプリケーションがそのユースケースに合わせて最適化できるように、解像度とフレームレートがどのように選択されるかについて説明します。

IVSCamera を IVSBroadcastSession にアタッチするときは、フレームレート `IVSVideoConfiguration.targetFramerate`、および解像度 `IVSVideoConfiguration.size` 向けにカメラが最適化されます。これらの値は、初期化時に `IVSBroadcastSession` に提供されます。

既知の問題と回避策

- のバグ `ReplayKit` により、ストリーミング中に有線ヘッドセットを接続すると、メモリが急増します。

回避方法: 有線ヘッドセットがすでに接続されている状態でストリームを開始する、Bluetooth ヘッドセットを使用する、あるいは外付けマイクは使用しないでください。

- `ReplayKit` ストリーム中にマイクを有効にしてからオーディオセッションを中断した場合 (例えば、電話や Siri のアクティブ化など)、システムオーディオは機能しなくなります。これは Apple で解決しようとしている `ReplayKit` バグです。

回避方法: 音声が入断した場合は、ブロードキャストを停止し、ユーザーに警告します。

- `AirPods AVAudioSession` カテゴリが に設定されている場合、 はオーディオを記録しません `record`。デフォルトでは、SDKは `playAndRecord` を使用するため、カテゴリが `record` に変更された場合にだけ、この問題が発生します。

回避策: オーディオの `AirPods` 録音に が使用される可能性がある場合は、アプリケーションがメディアを再生していない場合 `playAndRecord` でも を使用します。

- AirPods が iOS 12 デバイスに接続されている場合、他のマイクを使用してオーディオを録音することはできません。内部マイクに切り替えようとする、すぐにに戻ります AirPods。

回避策: 該当なし。AirPods が iOS 12 に接続されている場合、オーディオを録音できる唯一のデバイスです。

- (カスタムオーディオソースを使用して) 実際の速度よりも高速にオーディオデータを送信すると、オーディオドリフトが発生します。

回避方法: 実際の速度よりも高速にオーディオデータを送信しないでください。

- 高サンプルレート (44100 Hz 以上) と 2 つのチャンネルを使用すると、オーディオアーティファクトが 68 kbps 未満のビットレートで表示されることがあります。

回避方法: ビットレートを 68 kbps 以上に上げて、サンプルレートを 24000 Hz 以下に減らすか、チャンネルを 1 に設定します。

- IVSMicrophone デバイスでエコーキャンセレーションが有効になっている場合、`listAvailableInputSources` メソッドによって返されるマイクソースは 1 つだけです。

回避策: 該当なし。この動作は iOS によって制御されます。

- Bluetooth オーディオルートの変更は予測できない場合があります。セッション中に新しいデバイスを接続すると、iOS が入力ルートを自動的に変更する場合があります。また、同時に接続されている複数の Bluetooth ヘッドセットから選択することはできません。これは通常のブロードキャストとステージセッションの両方で起こります。

回避策: Bluetooth ヘッドセットを使用する場合は、ブロードキャストまたはステージを開始する前に接続し、セッション全体を通して接続したままにします。

- iOS では、ペアリングされた AirPods ケースを開いた後に AirPods ポップアップが表示され、そのケースに AirPods 自分自身を残すと、カメラへのアクセスが削除されます。その結果、ブロードキャストまたはステージの動画がフリーズします。

回避策: なし。iOS では、ポップアップのレンダリング中はカメラへのアクセスが完全に無効になるため、サードパーティのアプリケーションがポップアップを非表示にすることはできません。

IVS Broadcast SDK: ミキサーガイド (低レイテンシーストリーミング)

ミキサーは、複数の入力ソースを受け取り、単一の出力を生成するオーディオおよびビデオ処理ユニットです。これは、複数の画面上 (ビデオ) 要素とオーディオトラックを定義および管理できる強

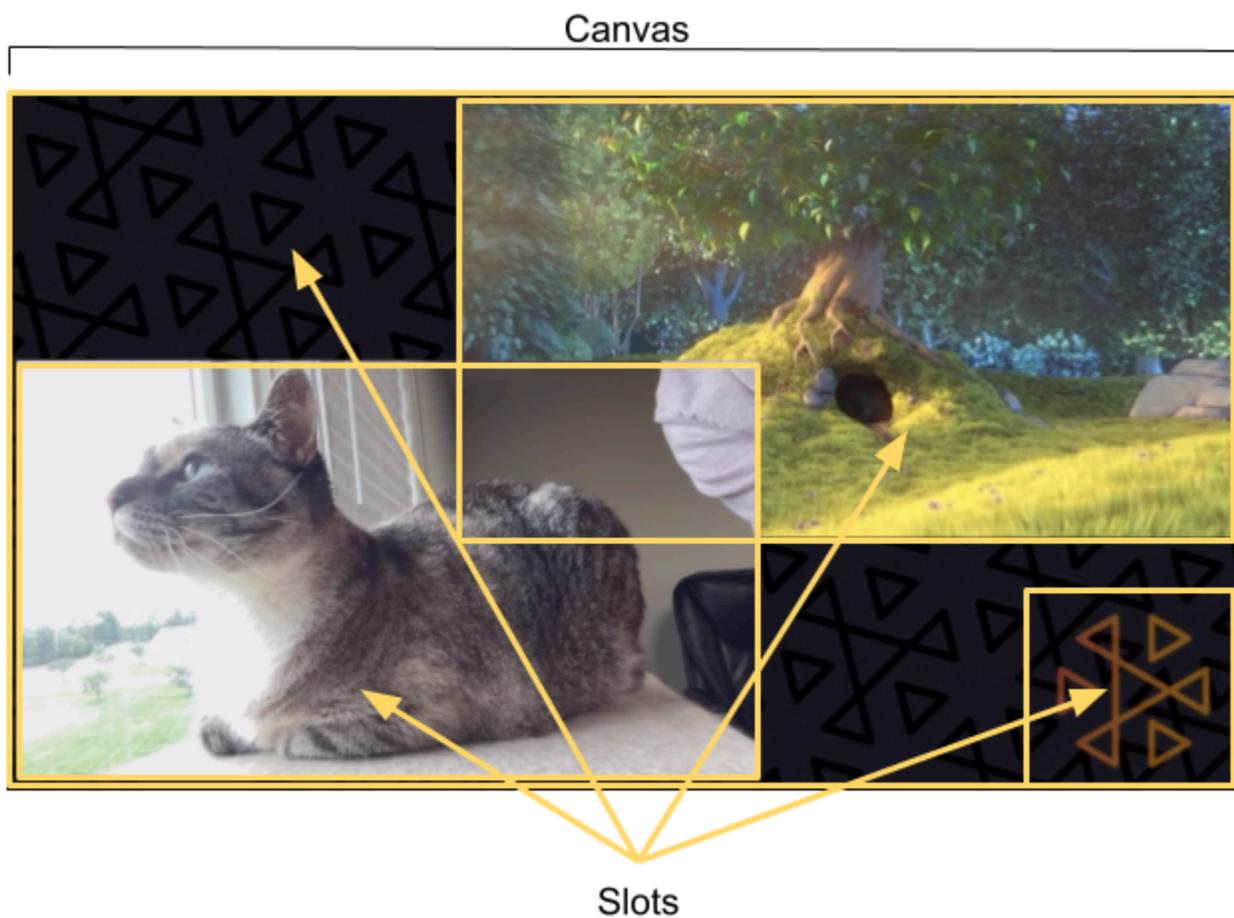
かな機能です。カメラ、マイク、スクリーンキャプチャ、アプリで生成されたオーディオとビデオなど、複数のソースからのビデオとオーディオを組み合わせることができます。トランジションを使用して、Amazon IVS にストリーミングする動画周辺にこれらのソースを移動させて、ストリーミングの途中でそれらを追加および削除することができます。

ミキサーにアクセスするには、以下を呼び出します。

Android の `BroadcastSession.getMixer()`

iOS の `IVSBroadcastSession.mixer`

用語



| 用語 | 説明 |
|---------|---|
| バインディング | 入力デバイスをスロットに関連付けるには、デバイスがバインドされたミキサー スロットに移動します。これは <code>Mixer.bind()</code> メソッドで行われます。スロットには、一度に1つのイメージ入力と1つのオーディオ入力をバインドで |

| 用語 | 説明 |
|-------------|--|
| | きます。Mixer.unbind() を呼び出して、スロットからデバイスをバインド解除できます。 |
| キャンバス | BroadcastSession 設定で定義されたビデオの表示範囲。キャンバスのサイズはビデオ設定と同じで、設定で指定したフレームレートと同じフレームレートで実行されます。 |
| デバイス | BroadcastSession へのオーディオまたはイメージ入力を生成するハードウェアまたはソフトウェアコンポーネント。デバイスの例としては、マイク、カメラ、Bluetooth ヘッドセット、画面キャプチャやカスタム画像入力などの仮想デバイスがあります。カスタム入力を除き、通常、デバイスオブジェクトへの参照を保持する必要はありません。代わりに、デバイスディスクリプタのコピーを保持できます。 |
| デバイスディスクリプタ | 入力デバイスに関する情報を含む構造。例えば、そのタイプ、システムアドレス、人間が読める「フレンドリー」な名前、モバイルデバイス上の物理的な位置など。この情報により、参照先デバイスを使用するかどうかを決定し、Amazon IVS がそのデバイスにアクセスできるようになります。 |
| スロット | <p>画面上のビジュアルエレメントの位置と、オーディオミックス内のオーディオトラックのプロパティを定義するコンテナです。ミキサーは 0 個以上のスロットで構成できます。スロットには、デバイスをバインドしてトランジションを実行するために使用できる文字列名が与えられます。上の図は、4 つのスロットを示しています。</p> <ul style="list-style-type: none">• 左下、カメラ入力あり• 右上にムービー入力あり• 右下に Amazon IVS のロゴが入っています• フルスクリーンのバックグラウンドイメージ <p>セッションを設定したら、addSlot および removeSlot ミキサーメソッドでスロットを追加および削除できます。</p> |

| 用語 | 説明 |
|---------|--|
| トランジション | <p>スロットを新しい位置に移動したり、そのプロパティの一部を変更するには、<code>Mixer.transition()</code> を使用します。このメソッドには次のものがあります。</p> <ul style="list-style-type: none"> スロットの次の状態を表す新しいスロット構造 ビデオのタイムラインを基準にして、アニメーションにかかる時間を指定するデューレーション。デューレーションを 0 に設定すると、ミックスされる次のフレームでトランジションが行われます。 アニメーションが完了したことを通知するオプションのコールバック。コールバックは、アニメーションを連鎖させるのに便利です。 |

キャンバスプロパティ

キャンバスのプロパティは、`BroadcastConfiguration` の作成時に指定した `BroadcastSession` に基づいて設定されます。Audio および Video 構造の複数のプロパティは、次のキャンバスに影響します。

| 名前 | 型 | 説明 |
|--------------------------------------|------------------------------|---|
| <code>Audio.channels</code> | 整数 | オーディオミキサーからの出力チャンネルの数。有効な値: 1、2。1 チャンネルはモノラルオーディオ、2 チャンネルはステレオオーディオです。デフォルト: 2。 |
| <code>Audio.sampleRate</code> | <code>AudioSampleRate</code> | オーディオミキサーからの 1 秒あたりのオーディオサンプル数。この値は、オーディオ信号の最高周波数の少なくとも 2 倍にする必要があります。人は 20 kHz 程度まで聞こえるので、一般的に 44.1 kHz と 48 kHz で十分です。デフォルト: 48 kHz です。 |
| <code>Video.defaultAspectMode</code> | <code>AspectMode</code> | <p>スロットのデフォルトのアスペクト比モード。有効な値:</p> <ul style="list-style-type: none"> <code>Fill</code> — 画像のアスペクト比を維持しますが、スロットを埋めます。必要に応じて、画像が切り取られます。 |

| 名前 | 型 | 説明 |
|-----------------------|------|---|
| | | <ul style="list-style-type: none"> • Fit — 画像のアスペクト比を維持しますが、画像全体をスロットに収めます。必要に応じて、スロットにレターボックスまたはピラーボックスが付いている場合があります。その値が設定されている場合、レター/ピラーボックスは fillColor になります。それ以外の場合は透明です (画像の背後にあるキャンバスの色が黒の場合、黒で表示される場合があります)。 • None — イメージのアスペクト比を維持しないでください。イメージは、スロットの寸法に合わせて拡大/縮小されます。 |
| Video.size | Vec2 | ビデオキャンバスのサイズ。 |
| Video.targetFrameRate | 整数 | キャンバスの 1 秒あたりのターゲットフレーム数。平均してこの値を満たす必要がありますが、特定の状況 (CPU 負荷が高い、ネットワーク輻輳など) でシステムがフレームをドロップする可能性があります。 |

スロットのプロパティ

スロットには、シーンをカスタマイズしてアニメーション化するために使用できる、いくつかの設定可能なプロパティがあります。Float または Vector の値は、0 秒より長いデユレーションのトランジションに対してリニア補間を使用してアニメートされます。

| 名前 | 型 | 説明 |
|--------|------------|---|
| aspect | AspectMode | <p>スロットでレンダリングされたイメージのアスペクト比モード。有効な値:</p> <ul style="list-style-type: none"> • Fill — 画像のアスペクト比を維持しますが、スロットを埋めます。必要に応じて、画像が切り取られます。 • Fit — 画像のアスペクト比を維持しますが、画像全体をスロットに収めます。必要に応じて、スロ |

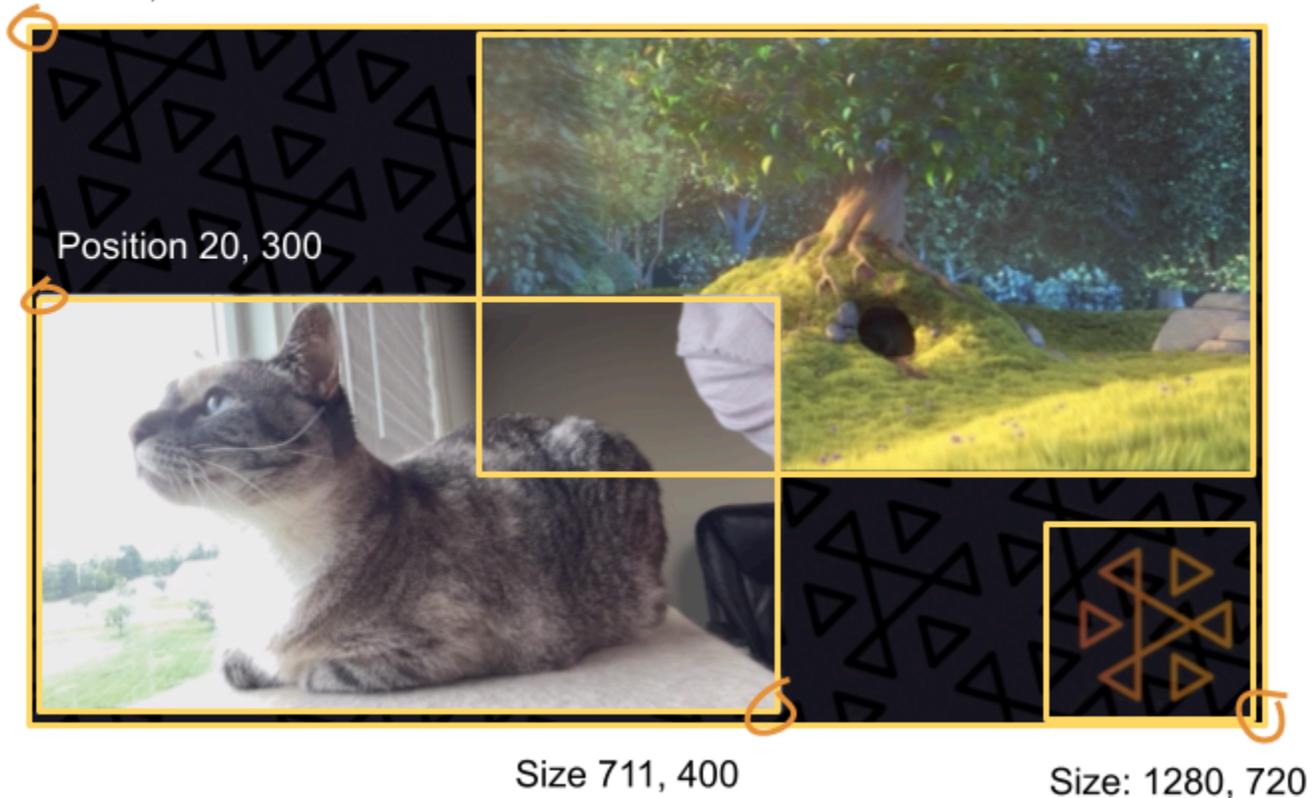
| 名前 | 型 | 説明 |
|------------------------------------|-------------------|--|
| | | <p>トにレターボックスまたはピラーボックスが付いている場合があります。その値が設定されている場合、レター/ピラーボックスは <code>fillColor</code> になります。それ以外の場合は透明です (画像の背後にあるキャンバスの色が黒の場合、黒で表示される場合があります)。</p> <ul style="list-style-type: none"> • None — イメージのアスペクト比を維持しないでください。イメージは、スロットの寸法に合わせて拡大/縮小されます。 <p>デフォルト: <code>matchCanvasAspectMode</code> が <code>true</code> の場合は、キャンバスの <code>aspect</code> と同じです。そうでない場合は、<code>Fill</code> です。</p> |
| <code>fillColor</code> | <code>Vec4</code> | <p>スロットと画像のアスペクト比が一致しない場合に、アスペクトフィットで使用する塗りつぶしカラー。フォーマットは (赤、緑、青、アルファ) です。有効な値 (チャンネルごとに): 0 - 1。デフォルト: (0, 0, 0, 0)。</p> |
| <code>gain</code> | 浮動小数点 | <p>オーディオゲイン。これは乗数であるため、1 を超える値であればゲインが増加し、1 より小さい値であれば減少します。有効な値は 0 ~ 2 です。デフォルト: 1。</p> |
| <code>matchCanvasAspectMode</code> | ブール値 | <p><code>true</code> の場合は、キャンバスの <code>Video.defaultAspectMode</code> 値を使用します。スロットの <code>aspect</code> プロパティを設定すると、これは <code>false</code> に設定されます。デフォルト: <code>true</code>。</p> |
| <code>matchCanvasSize</code> | ブール値 | <p><code>true</code> の場合、スロットのサイズはキャンバスのサイズと等しくなるように調整され、その位置は (0, 0) に設定されます。スロットの <code>size</code> プロパティを設定すると、これは <code>false</code> に設定されます。デフォルト: <code>true</code>。</p> |

| 名前 | 型 | 説明 |
|---------------------|------------|---|
| name | 文字列 | スロットの名前。これは、バインディングとトランジションのスロットを参照するために使用されます。デフォルト: "default"。 |
| position | Vec2 | キャンバスの左上隅からの相対で、スロット位置 (ピクセル) です。スロットの原点も左上です。 |
| preferredAudioInput | DeviceType | <p>優先するオーディオ入力デバイスタイプ。このスロットがバインドされておらず、指定されたタイプのオーディオデバイスがセッションに接続されている場合、デバイスはこのスロットに自動的にバインドされます。有効な値:</p> <ul style="list-style-type: none">• マイク — オンボードマイク、プラグ可能なヘッドフォン、Bluetooth ヘッドフォンなどのオーディオハードウェア。• システムオーディオ — オペレーティングシステムからキャプチャされたオーディオ。通常は画面録画を伴います。• ユーザーオーディオ — 作成したカスタムオーディオ入力。• Unknown — 優先デバイスはありません。スロットは常に手動でバインドされます。 |

| 名前 | 型 | 説明 |
|---------------------|------------|--|
| preferredVideoInput | DeviceType | <p>優先ビデオ入力デバイス。このスロットがバインドされておらず、指定されたタイプのビデオデバイスがセッションに接続されている場合、デバイスはこのスロットに自動的にバインドされます。有効な値:</p> <ul style="list-style-type: none"> カメラ — 前面カメラ、背面カメラ、広角カメラなどのオンボードカメラデバイス。 スクリーン — オペレーティングシステムからのスクリーンキャプチャ。 ユーザーイメージ — 作成したカスタムイメージおよびビデオ入力。 Unknown — 優先デバイスはありません。スロットは常に手動でバインドされます。 |
| size | Vec2 | <p>スロットのサイズ (ピクセル単位)。この値を設定すると、matchCanvasSize も false に設定されます。デフォルト: (0, 0); ただし、matchCanvasSize のデフォルトは true であるため、スロットのレンダリングサイズはキャンバスサイズであり、(0, 0) ではありません。</p> |
| transparency | 浮動小数点 | <p>スロットの透明度。これは、画像内のアルファ値との乗法です。不透明度は $1 - \text{transparency}$ です。有効な値: 0-1。0 は完全に不透明、1 は完全に透明です。デフォルト: 0。</p> |
| zIndex | 浮動小数点 | <p>スロットの相対的な順序。zIndex 値が高いスロットは、zIndex 値が低いスロットの上に描画されます。</p> |

ミキシング用のブロードキャストセッションの設定

Position 0, 0



ここでは、このガイドの冒頭にあるシーンに似たシーンを作成し、次の3つの画面上の要素を使用します。

- カメラの左下のスロット。
- ログオーバーレイ用の右下のスロット。
- 映画の右上のスロット。

キャンバスの原点は左上隅で、これはスロットでも同じであることに注意してください。したがって、スロットを (0, 0) に配置すると、スロット全体が見えるように左上隅に配置されます。

iOS

```
let config = IVSBroadcastConfiguration()
try config.video.setSize(CGSize(width: 1280, height: 720))
try config.video.setTargetFramerate(60)
config.video.enableTransparency = true
```

```
// Bottom Left
var cameraSlot = IVSMixerSlotConfiguration()
cameraSlot.size = CGSize(width: 320, height: 180)
cameraSlot.position = CGPoint(x: 20, y: 1280 - 200)
cameraSlot.preferredVideoInput = .camera
cameraSlot.preferredAudioInput = .microphone
cameraSlot.matchCanvasAspectMode = false
cameraSlot.zIndex = 2
try cameraSlot.setName("camera")

// Top Right
var streamSlot = IVSMixerSlotConfiguration()
streamSlot.size = CGSize(width: 640, height: 320)
streamSlot.position = CGPoint(x: 1280 - 660, y: 20)
streamSlot.preferredVideoInput = .userImage
streamSlot.preferredAudioInput = .userAudio
streamSlot.matchCanvasAspectMode = false
streamSlot.zIndex = 1
try streamSlot.setName("stream")

// Bottom Right
var logoSlot = IVSMixerSlotConfiguration()
logoSlot.size = CGSize(width: 320, height: 180)
logoSlot.position = CGPoint(x: 1280 - 340, y: 720 - 200)
logoSlot.preferredVideoInput = .userImage
logoSlot.preferredAudioInput = .unknown
logoSlot.matchCanvasAspectMode = false
logoSlot.zIndex = 3
try logoSlot.setTransparency(0.7)
try logoSlot.setName("logo")

config.mixer.slots = [ cameraSlot, streamSlot, logoSlot ]
```

Android

```
// Bottom Left
val cameraSlot = BroadcastConfiguration.Mixer.Slot.with { s ->
    s.setSize(320, 180)
    s.position = BroadcastConfiguration.Vec2(20, 1280 - 200)
    s.preferredVideoInput = Device.Descriptor.DeviceType.CAMERA
    s.preferredAudioInput = Device.Descriptor.DeviceType.MICROPHONE
    s.matchCanvasAspectMode = false
    s.zIndex = 2
```

```
s.name = "camera"
s
}

// Top Right
val streamSlot = BroadcastConfiguration.Mixer.Slot.with { s ->
    s.setSize(640, 320)
    s.position = BroadcastConfiguration.Vec2(1280 - 660, 20)
    s.preferredVideoInput = Device.Descriptor.DeviceType.USER_IMAGE
    s.preferredAudioInput = Device.Descriptor.DeviceType.USER_AUDIO
    s.matchCanvasAspectMode = false
    s.zIndex = 1
    s.name = "stream"
    s
}

// Bottom Right
val logoSlot = BroadcastConfiguration.Mixer.Slot.with { s ->
    s.setSize(320, 180)
    s.position = BroadcastConfiguration.Vec2(1280 - 340, 720 - 200)
    s.preferredVideoInput = Device.Descriptor.DeviceType.USER_IMAGE
    s.preferredAudioInput = Device.Descriptor.DeviceType.UNKNOWN
    s.matchCanvasAspectMode = false
    s.zIndex = 3
    s.name = "logo"
    s.transparency = 0.7
    s
}

val config = BroadcastConfiguration.with { c ->
    c.mixer.slots = listOf(cameraSlot, streamSlot, logoSlot)
    c.video.targetFramerate = 60
    c.video.setSize(1280, 720)
    c
}
```

スロットを追加する

構成で `BroadcastSession` を作成したら、ミキサーにスロットを追加したり、ミキサーからスロットを削除したりできます。ここでは、イメージ用の大きな背景スロットをミキサーに追加します。

iOS

```
// Background. We will use most of the defaults for this slot.
var backgroundSlot = IVSMixerSlotConfiguration()
backgroundSlot.preferredVideoInput = .userImage
backgroundSlot.preferredAudioInput = .unknown
backgroundSlot.matchCanvasAspectMode = false
try backgroundSlot.setName("background")

session.mixer.addSlot(backgroundSlot)
```

Android

```
// Background. We will use most of the defaults for this slot.
val backgroundSlot = BroadcastConfiguration.Mixer.Slot.with { s ->
    s.preferredVideoInput = Device.Descriptor.DeviceType.USER_IMAGE
    s.preferredAudioInput = Device.Descriptor.DeviceType.UNKNOWN
    s.matchCanvasAspectMode = false
    s.name = "background"
}

session.mixer.addSlot(backgroundSlot)
```

スロットの削除

スロットを削除するには、削除するスロットの名前で `BroadcastSession.Mixer.removeSlot` を呼び出します。そのスロットにバインドされているデバイスはすべて自動的にバインド解除されるため、引き続き使用する場合は、別のスロットに再バインドする必要があります。

トランジションのあるアニメーション

ミキサートランジション方式は、スロットの構成を新しい構成に置き換えます。この置換は、デューレーションを 0 より高く秒単位で設定することで、時間の経過とともにアニメートできます。

アニメーション化できるプロパティ

スロット構造のすべてのプロパティをアニメートできるわけではありません。Float タイプに基づくすべてのプロパティをアニメートできます。その他のプロパティは、アニメーションの開始時または終了時に有効になります。

| 名前 | アニメ化可否 | インパクトポイント |
|---|--------|--------------|
| aspect | No | End |
| fillColor | Yes | Interpolated |
| gain | Yes | Interpolated |
| matchCanvasAspectMode | No | スタート |
| matchCanvasSize | No | スタート |
| name | No | 該当なし |
| 注意:スロットの名前は変更できません。 | | |
| position | Yes | Interpolated |
| preferredAudioInput | No | End |
| preferredVideoInput | No | End |
| size | Yes | Interpolated |
| transparency | Yes | Interpolated |
| zIndex | Yes | 不明 |
| 注: zIndex は 2D 平面を 3D 空間内で移動するため、アニメーションの途中で 2 つの平面が交差したときにトランジションが発生します。これは計算できますが、開始A値と終了 zIndex 値によって異なります。トランジションをよりスムーズにするために、これを transparency と組み合わせます。 | | |

シンプルな例

以下は、[ミキシング用のブロードキャストセッションの設定](#)で上記で定義された構成を使用したフルスクリーンカメラテイクオーバーの例です。これは 0.5 秒でアニメーション化されます。

iOS

```
// Bottom Left
var bigCameraSlot = cameraSlot
bigCameraSlot.size = CGSize(width: 1280, height: 720)
bigCameraSlot.position = CGPoint(x: 0, y: 0)

session.mixer.transition("camera", bigCameraSlot, 0.5) {
    println("animation completed!")
}
```

Android

```
// Bottom Left
val bigCameraSlot = cameraSlot.changing { s ->
    s.setSize(1280, 720)
    s.position = BroadcastConfiguration.Vec2(0, 0)
    s
}

session.mixer.transition("camera", bigCameraSlot, 0.5) {
    print("animation completed!")
}
```

ブロードキャストのミラーリング

| | |
|---|------------------|
| ブロードキャストでアタッチされた画像デバイスを以下の方向にミラーリングする場合 | 以下の条件には負の値を使用します |
| 水平方向 | スロットの幅 |
| 垂直方向 | スロットの高さ |
| 水平方向と垂直方向の両方 | スロットの幅と高さの両方 |

ミラーリング時にスロットを正しい位置に配置するには、同じ値で位置を調整する必要があります。

以下は、ブロードキャストを水平方向と垂直方向にミラーリングする例です。

iOS

水平ミラーリング:

```
var cameraSlot = IVSMixerSlotConfiguration
cameraSlot.size = CGSize(width: -320, height: 720)
// Add 320 to position x since our width is -320
cameraSlot.position = CGPoint(x: 320, y: 0)
```

垂直ミラーリング:

```
var cameraSlot = IVSMixerSlotConfiguration
cameraSlot.size = CGSize(width: 320, height: -720)
// Add 720 to position y since our height is -720
cameraSlot.position = CGPoint(x: 0, y: 720)
```

Android

水平ミラーリング:

```
cameraSlot = BroadcastConfiguration.Mixer.Slot.with {
    it.size = BroadcastConfiguration.Vec2(-320f, 180f)
    // Add 320f to position x since our width is -320f
    it.position = BroadcastConfiguration.Vec2(320f, 0f)
    return@with it
}
```

垂直ミラーリング:

```
cameraSlot = BroadcastConfiguration.Mixer.Slot.with {
    it.size = BroadcastConfiguration.Vec2(320f, -180f)
    // Add 180f to position y since our height is -180f
    it.position = BroadcastConfiguration.Vec2(0f, 180f)
    return@with it
}
```

注: このミラーリングは `ImagePreviewView` (Android) や `IVSImagePreviewView` (iOS) の `setMirrored` メソッドとは異なります。このメソッドはデバイス上のローカルプレビューにのみ影響し、ブロードキャストには影響しません。

IVS Broadcast SDK: カスタムイメージソース (低レイテンシーストリーミング)

このガイドは、ブロードキャストセッション ([Android](#)、[iOS](#)) のセットアップ方法と [ミキサー API の使用方法](#) に既に精通していることを前提としています。

カスタム画像入力ソースを使用することで、プリセットカメラやスクリーン共有に限定されるのではなく、アプリケーションが独自の画像入力をブロードキャスト SDK に提供できます。カスタム画像ソースは、半透明の透かしや静的な「be right back」(すぐに戻ります) シーンと同じくらいシンプルにすることも、アプリケーションがカメラにビューティーフィルターを追加するなど、追加のカスタム処理を実行できるようにすることもできます。

透かしやビューティーフィルターを備えたカメラなど、複数のカスタム画像ソースを使用できます。カメラのカスタムコントロール (カメラアクセスを必要とするビューティーフィルターライブラリを使用するなど) のためにカスタム画像入力ソースを使用する場合、ブロードキャスト SDK はカメラの管理に責任を負わなくなります。代わりに、アプリケーションは、カメラのライフサイクルの正しい処理に責任を負うようになります。アプリケーションがカメラをどのように管理すべきかについては、プラットフォームの公式ドキュメントを参照してください。

Android

ブロードキャストセッションを作成したら、画像入力ソースを作成します。

```
SurfaceSource surfaceSource = broadcastSession.createImageInputSource();
```

このメソッドは、標準の Android [Surface](#) に基づく画像ソースである `SurfaceSource` を返します。ブロードキャストセッションに自動的にアタッチされるため、後で `attachDevice(...)` メソッドを使用する必要はありません。ただし、`SurfaceSource` はスロットにバインドする必要があります。これについては、後で説明します。`SurfaceSource` のサイズを変更したり、回転したりできます。`ImagePreviewView` を作成して、その内容のプレビューを表示することもできます。

基盤の `Surface` を取得するには:

```
Surface surface = surfaceSource.getInputSurface();
```

この `Surface` は、`Camera2`、`OpenGL ES`、他のライブラリなどの画像プロデューサーの出力バッファとして使用できます。最も簡単なユースケースは、静的なビットマップまたは色を `Surface` のキャンバスに直接描画することです。ただし、多くのライブラリ (ビューティーフィルターライブラ

りなど)は、アプリケーションがレンダリングのために外部の Surface を指定できるようにするメソッドを提供します。このようなメソッドを使用して、この Surface をフィルターライブラリに渡すことができます。これにより、ライブラリは、ブロードキャストセッションのために処理されたフレームを出力してストリーミングできます。

最後に、SurfaceSource は、ブロードキャストセッションによってストリーミングされるためには、Mixer.Slot にバインドされる必要があります。

```
broadcastSession.getMixer().bind(surfaceSource, "customSlot");
```

[Android サンプルコード](#)には、さまざまな方法でカスタム画像ソースを使用するいくつかの例があります。

- 半透明の透かしが MixerActivity に追加されます。
- MP4 ファイルは MixerActivity でループされます。
- [CameraManager](#) ユーティリティクラスは、CustomActivity の Camera2 メソッドを使用してデバイスのカメラのカスタム管理を行います。これは、単純なセピアフィルターを適用します。この例は、カメラを管理し、ブロードキャストセッションのカスタム SurfaceSource をカメラキャプチャリクエストに渡す方法を示しているため、特に役立ちます。他の外部ライブラリを使用する場合は、ブロードキャストセッションによって提供される Android Surface に出力するようにライブラリを設定する方法に関するドキュメントに従ってください。

iOS

ブロードキャストセッションを作成したら、画像入力ソースを作成します。

```
let customSource = broadcastSession.createImageSource(withName: "customSourceName")
```

このメソッドは、アプリケーションが手動で CMSampleBuffers を送信できるようにする画像ソースである IVSCustomImageSource を返します。サポートされているピクセル形式については、「iOS ブロードキャスト SDK リファレンス」を参照してください。最新バージョンへのリンクは、最新のブロードキャスト SDK リリースの「[Amazon IVS リリースノート](#)」にあります。ソースはブロードキャストセッションに自動的にアタッチされないため、ソースがストリーミングされる前に、画像ソースをセッションにアタッチし、スロットにバインドする必要があります。

```
broadcastSession.attach(customSource, toSlotWithName: "customSourceSlot", onComplete: nil)
```

カスタムソースがアタッチされてバインドされた後、アプリケーションはカスタムソースに直接 CMSampleBuffers を送信できます。onComplete コールバックを使用して開始することもできます。

カスタムソースに送信されたサンプルは、ブロードキャストセッションでストリーミングされます。

```
customSource.onSampleBuffer(sampleBuffer)
```

動画のストリーミングには、このメソッドをコールバックで使用します。例えば、カメラを使用している場合、AVCaptureSession から新しいサンプルバッファを受信するたびに、アプリケーションはサンプルバッファをカスタム画像ソースに転送できます。必要に応じて、カスタム画像ソースにサンプルを送信する前に、アプリケーションでさらなる処理 (ビューティーフィルターなど) を適用できます。

静的画像については、カスタム画像ソースのスロットバインディングが変更された場合、またはソースがデタッチされてブロードキャストセッションに再アタッチされた場合、最初のサンプルの後、アプリケーションはサンプルを再送信する必要があります。例えば、スロットをミキサーから削除してから、そのミキサーにスロットを追加する場合は、サンプルを再送信する必要があります。

[iOS サンプルアプリケーション](#)には、さまざまな方法でカスタム画像ソースを使用するいくつかの例があります。

- 半透明の透かしが MixerViewController に追加されます。
- MP4 ファイルは MixerViewController でループされます。
- デバイスのカメラによる CIFilter 実装は、CustomSourcesViewController に追加されました。これにより、アプリケーションは Amazon IVS Broadcast SDK とは独立してデバイスのカメラを管理できます。AVCaptureSession を使用してデバイスのカメラから画像をキャプチャし、CIFilter 実装を使用して画像を処理し、ライブストリーミング用に CMSampleBuffers を customSource に送信します。

Amazon IVS Player SDK

Amazon Interactive Video Service (IVS) を使用するには、Amazon IVS プレイヤーが必要です。このプレイヤーは、Amazon IVS ストリームを再生するためのクロスプラットフォームの SDK スイートです。Amazon IVS アーキテクチャを活用するように設計され、Amazon IVS の再生に最適化されています。

当社が性能を保証できる唯一のプレイヤーが Amazon IVS プレイヤーです。低レイテンシーを実現するには、Amazon IVS プレイヤーが必要になります。

Amazon IVS プレイヤーの主な機能をご紹介します。

- **Ultra-low-latency ストリーミング** — 低レイテンシーは、視聴者エクスペリエンスを強化する優れたインタラクティブユーザーエクスペリエンスを構築する上で重要なコンポーネントです。レイテンシーは、配信者と視聴者の間の伝送経路全体に段階的に入り込み、応答性を低下させます。

End-to-end レイテンシーは、ライブストリームがカメラにキャプチャされてから視聴者の画面に表示されるまでの遅延です。Amazon IVS は、超低 end-to-end レイテンシー (ブロードキャストの場所とブロードキャスターの設定に応じて 5 秒未満) を実現するように設計されています。低レイテンシーを実現するには、Amazon IVS プレイヤーが必要です。
- **クロスプラットフォームの一貫性** — 視聴者はさまざまなプラットフォームで配信動画を視聴します。モバイルデバイスからウェブブラウザまで、Amazon IVS プレイヤーはすべての視聴者に同様の体験を提供します。この一貫性は、すべてのプラットフォームに同じプレイヤー機能のライブラリを使わせることで可能になります。このプレイヤーライブラリは、Amazon IVS アーキテクチャにとって欠かせない要素です。1つの動画スタックを使用することで、低レイテンシーモード、タイムドメタデータ、分析、エラー追跡、レポート、ログ記録など、動画再生のすべての動作を、サポート対象のすべてのプラットフォームで一貫した方法で利用できるようにしています。
- **アダプティブビットレートストリーミング (ABR)** — Amazon IVS プレイヤーは、低レイテンシー環境に最適化された ABR アルゴリズムを使用しています。サービスの品質と帯域幅の可用性をリアルタイムで測定し、動画の品質とバッファレベルとを調整することで、中断なく再生を行います。接続品質が低下すると、ABR が低ビットレートに切り替わり、接続品質が改善すると、より高いビットレートに切り替わります。
- **タイムドメタデータ** — Amazon IVS プレイヤーはタイムドメタデータをサポートしています。これは、投票やクイズなどインタラクティブな要素の構築に使用できます。メタデータとは、他のデータに関する情報を記述して付与するデータのセットのことです。「タイムド」メタデータでは、ストリームに関するデータにタイムコードが付随します。このタイムコードは、動画の再生中

に、例えば次のように、データに基づいてアクションをトリガーするキューポイントとして機能します。

- スポーツのストリームにおける選手の統計情報の送信
- ショッピングのライブストリームにおける商品情報の送信
- クイズのライブストリームにおける質問の送信
- 堅牢なエラー処理 — 一時的なエラーに対処することで視聴体験の中断を適切に回避します。Amazon IVS プレイヤーの堅牢なエラー処理は、ストリーミング中の多くの潜在的エラーを検出し、代替の表示に自動的に切り替えます。視聴者は、中断なく継続して配信動画を視聴でき、自分で何らかの修正措置を講じる必要がありません。
- 統合が容易 — Amazon IVS プレイヤー API は、IVS のお客様のアプリケーションとプレイヤーライブラリとのギャップを埋めます。API には、サポートされているすべてのプラットフォームのバイインディングがあり、使い慣れたコーディング環境とテクニックを使用しながら、プレイヤーをアプリケーションに容易に統合できるようにしています。UI の要素をフルにコントロールでき、お客様は、アプリケーションのブランディングや外観面をカスタマイズできます。

Amazon IVS プレイヤーは Airplay でのキャストをサポートしていません。Chromecast によるキャストは、デフォルトの Chromecast レシーバーアプリケーションを使用して、プレイヤー外部に実装できます。ただし、これらのアプリケーションのレイテンシーが Amazon IVS Player SDK よりも高いため、シームレスに切り替えることはできません。Amazon IVS Broadcast SDK: 「[低レイテンシーストリーミング](#)」および「[リアルタイムストリーミング](#)」に関するドキュメントも参照してください。

ブラウザとプラットフォームの要件

各種ブラウザの最新リリース版の詳細については、以下を参照してください。

- [Chrome プラットフォームのステータス](#)
- [Firefox のリリース](#)
- [Microsoft Edge のリリーススケジュール](#)
- [Safari リリースノート](#)

Amazon IVS が一部の古いブラウザでも動作する場合がありますが、古いブラウザに関連するバグは修正されません。

IVS Player Web SDK (Video.js および Player JW 統合を含む) は、ブラウザのような環境ではサポートされていません。これには、ウェブアプリケーションをサポートするネイティブデバイス WebViews と「10 フィートデバイスTVs、コンソール、セットトップボックス) が含まれます。以下の表以外の特定のブラウザのサポートが不明な場合は、IVS サポートにお問い合わせください。

デスクトップブラウザ

| デスクトップブラウザ | サポートされているプラットフォーム | サポートされているバージョン |
|------------|-------------------|--|
| Chrome | Windows、macOS | 2 つのメジャーバージョン (最新バージョンと 1 つ前のバージョン) |
| Firefox | Windows、macOS | 2 つのメジャーバージョン (最新バージョンと 1 つ前のバージョン) |
| Edge | (Windows 8.1 以降) | 44.0 以降

(Microsoft Edge Legacy の自動品質モードでは、標準レイテンシーによる再生のみがサポートされ、超低レイテンシー再生はサポートされていません。自動品質モードは、ABR が有効かどうかを示します。例えば Web プレイヤーの場合は、 <code>setAutoQualityMode</code> を参照してください。) |
| Safari | macOS | 2 つのメジャーバージョン (最新バージョンと 1 つ前のバージョン)

(macOS 14 以降の Safari の自動品質モードでは、IVS プレイヤー 1.3.0 以降は超低レイテンシー再生をサポートしています。以前のバージョンの Safari および IVS プレイヤーは通常のレイテンシー再生のみをサポートしています。上記の「自動品質モード」を参照してください。) |

モバイルブラウザ

| モバイルブラウザ | サポートされているバージョン |
|-----------------------------------|---|
| Chrome for iOS、Safari for iOS | 2つのメジャーバージョン (最新バージョンと1つ前のバージョン)

(超低レイテンシー再生はサポートされていません。標準レイテンシーによる再生はサポートされています。この制限は、iOSのすべてのブラウザに適用されます)。

(時間指定メタデータは Player 1.3.0 以降のみでサポートされています)。 |
| iPadOS 向け Chrome、iPadOS 向け Safari | 2つのメジャーバージョン (最新バージョンと1つ前のバージョン)

(「Request Mobile Website」が選択されている場合:
<ul style="list-style-type: none"> 超低レイテンシー再生はサポートされていません。 (時間指定メタデータは Player 1.3.0 以降のみでサポートされています)。 |
| Chrome for Android | 2つのメジャーバージョン (最新バージョンと1つ前のバージョン) |

ネイティブプラットフォーム

| プラットフォーム | サポートされるバージョン | サポートされるデバイス |
|----------|-------------------|-------------|
| Android | 5.0 (Lollipop) 以降 | 携帯電話とタブレット |
| iOS | 12.0 以降 | すべて |

IVS は、少なくとも 4 つの iOS メジャーバージョンと 6 つの Android メジャーバージョンをサポートしています。現在のサポート対象バージョンは、これらの最小値よりも多い可能性があります。メジャーバージョンがサポートされなくなる場合は、少なくとも 3 か月前に SDK リリースノートでお客様にお知らせします。

サードパーティ製プレイヤーのレイテンシーの削減

Basic および Standard チャンネルタイプの場合: レイテンシーを最小限に抑えるには、Amazon IVS プレイヤーを使用する必要があります。サードパーティ製プレイヤー (iOS Safari など) では、次の設定をすることでレイテンシーを約 10 秒に短縮できます。

- エンコーダ (OBS など) のキーフレーム間隔を 2 秒以下に設定します。
- ?keyframeInterval=2 を RTMP(S) URL に追加します。例 : `rtmps://a1b2c3d4e5f6.global-contribute.live-video.net:443/app/sk_us-west-2_abcd1234efgh5678ijkl?keyframeInterval=2`

注: RTMP URL の一部として指定するキーフレーム間隔は、エンコーダで設定された値以上でなければなりません。そうでない場合、再生で問題が発生する可能性があります。値は 2~6 の整数を任意に設定できますが、2 に設定するとレイテンシーが最も低くなります。

Advanced チャンネルタイプの場合: 上記のガイダンスは適用されません。Advanced チャンネルタイプは、エンコーディングの効率性のためにキーフレーム間隔を自動的に生成します。キーフレーム間隔は、ソースエンコーディングキーフレームの間隔設定を問わず、最大 2 秒です。

iOS Safari

iOS Safari では、IVS プレイヤーでサービスワーカーを使用するように設定し、これを使用することでレイテンシーを約 6~8 秒に削減することができます。実装の詳細とサンプルレファレンスについては、「Player SDK: Web ガイド」の「[サービスワーカーのセットアップ](#)」を参照してください。

注: レイテンシーを最小限に抑えるには、IVS ストリームでキーフレーム間隔を 2 秒に設定する必要があります。

オーディオのみの再生

すべての IVS チャンネルタイプは、オーディオのみのレンディションをサポートします。これはモバイルアプリケーションにとって特に価値があります。例えば、モバイルアプリでは、ユーザーがアプリケーションを背景にして帯域幅を節約するときに、プレイヤーをオーディオのみのレンディションに切り替えることができます。

ADVANCED-SD および ADVANCED-HD チャンネルの場合、オーディオのみのレンディションはマルチバリエーションプレイリストに自動的に含まれます。BASIC チャンネルと STANDARD チャンネルの場合

合、オーディオのみのレンディションを含めるには、再生 URL に `?allow_audio_only=true` クエリパラメータを追加する必要があります。

注: IVS ウェブプレイヤー SDK は、バージョン 1.24.0 以降でのみオーディオのみの再生をサポートします。

サポート

ストリームに再生エラーやその他再生に関する問題が発生した場合は、プレイヤー API を介して再生セッションの一意の識別子を特定します。

| 次の Amazon IVS プレイヤーでは | 以下を使用 |
|-----------------------|---|
| Android | <code>sessionId</code> 関数 |
| iOS | <code>IVSPlayer</code> の <code>sessionId</code> プロパティ |
| Web | <code>getSessionId</code> 関数 |

この再生セッション識別子を AWS サポートに伝えます。それにより、サポートはトラブルシューティングに役立つ情報を入手できます。

注: プレイヤーは継続的に改良されています。利用可能なバージョンと修正済みの問題については [Amazon IVS リリースノート](#) を参照してください。必要に応じて、サポートに連絡する前にお使いのプレイヤーのバージョンを更新し、問題が解決するかどうかを確認してください。

バージョンニング

Amazon IVS Player SDK は、[セマンティックバージョンニング](#) を使用しています。

以下の解説は、次を前提としています。

- 最新リリースは 4.1.3。
- 1 つ前のメジャーバージョンの最新リリースは 3.2.4。
- バージョン 1.x の最新リリースは 1.5.6。

最新バージョンのマイナーリリースとして、下位互換性のある新機能が追加されています。この場合、次回の新機能のセットは、バージョン 4.2.0 として追加されます。

下位互換性のあるマイナーなバグ修正が、最新バージョンのパッチリリースとして追加されています。ここでは、次回のマイナーなバグ修正のセットは、バージョン 4.1.4 として追加されます。

下位互換性のあるメジャーなバグ修正は異なる方法で処理されます。これらはいくつかのバージョンに追加されています。

- 最新バージョンのパッチリリース。こちらは、バージョン 4.1.4 です。
- 1 つ前のマイナーバージョンのパッチリリース。こちらは、バージョン 3.2.5 です。
- 最新バージョン 1.x リリースのパッチリリース。こちらは、バージョン 1.5.7 です。

メジャーなバグ修正は、Amazon IVS 製品チームによって定義されています。典型的な例に、重要なセキュリティ更新のほか、お客様に必要な選別された修正があります。

注: 上記の例では、リリースされたバージョンの数字は、連番でインクリメントされています(4.1.3 → 4.1.4、など)。実際は、1 つ以上のパッチ番号が内部に残り、リリースされないままになることもあります。そのため、リリースされたバージョンは 4.1.3 から (例えば) 4.1.6 に増えることもあります。

Amazon IVS Player SDK: Web のガイド

Amazon Interactive Video Service (IVS) Player SDK for Web は、Video.js のような [プレイヤーフレームワーク](#) と統合するか、HTML の <video> 要素の上でスタンドアロンで使用することができます。

Web プレイヤーの最新バージョン : 1.24.0 ([リリースノート](#))

リファレンスドキュメント : Amazon IVS Web プレイヤーで利用できる最も重要な方法については、<https://aws.github.io/amazon-ivs-player-docs/1.24.0/web/> のリファレンスドキュメントを参照してください。

開始方法

当社は、script タグ、ならびに npm モジュールを介してサポートを提供しています。

デモ

次のライブデモでは、この Web プレイヤーを、Content Delivery Network の script タグを用いて使用する方法を紹介しています。 [Amazon IVS Player Sample](#)

スクリプトタグを使ってセットアップする

script タグを使用して Amazon IVS プレイヤーを設定する

1. 次のタグを含めます (最新バージョンのプレイヤーの場合)。

```
<script src="https://player.live-video.net/1.24.0/amazon-ivs-player.min.js">
```

2. amazon-ivs-player.min.js がロードされると、IVSPlayer 変数がグローバルコンテキストに追加されます。これは、プレイヤーインスタンスの作成に使用するライブラリです。まず、isPlayerSupported をチェックして、ブラウザが IVS プレイヤーをサポートしているか確認します。

```
if (IVSPlayer.isPlayerSupported) { ... }
```

次に、プレイヤーインスタンスを作成するために、IVSPlayer オブジェクトで create 関数を呼び出します。

```
const player = IVSPlayer.create();
```

Amazon IVS Player SDK for Web は、ウェブワーカーを使用して動画の再生を最適化します。

3. プレイヤーインスタンスで load および play 関数を使用してストリームをロードし、再生します。

```
player.load("PLAYBACK_URL");  
player.play();
```

この場合、PLAYBACK_URL は、ストリームキーがリクエストされたときに Amazon IVS API から返される URL です。

サンプルコード

この例では、PLAYBACK_URL をロードしたいソースストリームの URL に置き換えてください。この例では、Amazon IVS プレイヤーの最新バージョンを使用しています。

```
<script src="https://player.live-video.net/1.24.0/amazon-ivs-player.min.js"></script>  
<video id="video-player" playsinline></video>  
<script>
```

```
if (IVSPlayer.isPlayerSupported) {
  const player = IVSPlayer.create();
  player.attachHTMLVideoElement(document.getElementById('video-player'));
  player.load("PLAYBACK_URL");
  player.play();
}
</script>
```

<video> タグでは、iOS Safari でのインライン再生に `playsinline` が必要です。 <https://webkit.org/blog/6784/new-video-policies-for-ios/> を参照してください。

NPM でセットアップする

Webpack 設定ファイルの例を含むガイダンスについては、次のリポジトリを参照してください: <https://github.com/aws-samples/amazon-ivs-player-web-sample>。

注: 独自のドメインからプレイヤーの静的アセットをホストする場合は、バイナリ (`amazon-ivs-wasmworker.min.wasm`) の「Content-Type WebAssembly」レスポンスヘッダーを「application/wasm」に設定する必要があります。また、アセットを gzip して、ワイヤ経由でダウンロードされるバイト数を減らし、プレイヤーの再生開始時間を短縮する必要があります。

TypeScript

を使用している場合 TypeScript、npm パッケージにはインポートして使用できるタイプが含まれます。これらのタイプに関する詳細については、「[Amazon IVS Player SDK: Web Reference](#)」を参照してください。

サービスワーカーの設定

ネイティブの再生機能のみをサポートするブラウザ (主として iOS Safari) で再生する場合は、サービスワーカーをセットアップして構成することで、レイテンシーをさらに低くすることができます。詳細については、「[サードパーティ製プレイヤーのレイテンシーの削減](#)」を参照してください。

サービスワーカーを使用するために Amazon IVS プレイヤーをセットアップするには

1. CDN からの IVS サービスワーカーの読み込先となるファイルを作成します。サービスワーカーと、それを取得するページは、同じドメインでホストされている必要があるため、このファイルが必要です。

`amazon-ivs-service-worker-loader.js` または類似の名前でファイルを作成し、次の行を追加します。

```
importScripts('https://player.live-video.net/1.24.0/amazon-ivs-service-worker.min.js');
```

2. プレイヤーインスタンスを作成するときに、次の `amazon-ivs-service-worker-loader.js` ファイルを参照する `serviceWorker` 設定を渡します。

```
const player = IVSPlayerPackage.create({
  serviceWorker: {
    url: 'amazon-ivs-service-worker-loader.js'
  }
});
```

3. `video` 要素で、`crossOrigin` 属性に `anonymous` を設定します。これは、サービスワーカーがマニフェストを変更できるようにするために必要です。

注: サービスワーカーをローカルでテストするには、ページを `localhost` または `https` から提供する必要があります。

ライブデモについては、以下のリポジトリにあるサービスワーカーの例を参照してください。

<https://github.com/aws-samples/amazon-ivs-player-web> サンプル

オーディオのみの再生

オーディオのみの品質は、`setQuality()`メソッドで手動で選択する必要があります。プレイヤーは2番目の引数 `true`の値をサポートしていないため `adaptive`、この引数はデフォルトであることに注意してください `false`。

再生を開始する前に品質をオーディオのみに設定するには、`READY`イベント `setQuality()`内で を呼び出します。

```
player.addEventListener(PlayerState.READY, () => {
  const qualities = player.getQualities();
  const audioOnly = qualities.find(q => q.name === 'audio_only');
  if (audioOnly) {
    player.setQuality(audioOnly);
  }
});
```

内で品質を設定することは、自動再生モードと非自動再生モードの両方で `READY`機能します。

フレームワークの統合

Amazon IVS Player SDK for Web は、ユーザーが選択したフレームワークと簡単に統合できるように設計されています。公式の Video.js 統合 (Video.js の用語では「tech」) を提供しています。

以下は、当社が提供している Web プレイヤーの簡単な比較表です。

| プレイヤータイプ | 説明 | UI | プラグイン |
|--|--|--|--|
| Amazon IVS Player SDK for Web | より高度な制御を必要とする開発者向けの軽量でカスタマイズ可能なオプション。 | いいえ | いいえ |
| Amazon IVS Player Tech for Video.js | フル機能のオプション。既に Video.js を使用していて、ターンキーソリューションが必要な場合に適しています。 | はい
(Video.js Skins) | はい
(Video.js Plugins) |
| JW Player 用 Amazon IVS Player プロバイダー | フル機能のオプション。既に JW Player を使用していて、ターンキーソリューションが必要な場合に適しています。 | はい | 該当なし |

コンテンツセキュリティポリシーの操作

Amazon IVS Web Player SDK は、コンテンツセキュリティポリシー (CSP) を使用するページで動作するように設定されています。主要な CSP ディレクティブをいくつか設定する必要があります。ここでは、必要最小限のディレクティブのセットについて説明します。特定の設定によっては、追加のディレクティブとソースが必要になる場合があります。

次のディレクティブは、CSP に最低限必要なものです。

```
worker-src blob;;
media-src blob;;
connect-src *.live-video.net;
script-src 'wasm-unsafe-eval';
```

注: 古いバージョンのブラウザでは、上記の CSP ルール (wasm-unsafe-eval など) を 1 つ以上認識せず、かわりにきわめて緩やかな CSP ポリシー (unsafe-eval) が必要になる可能性があります。ただし、これは CSP の全ポイントに対して機能し、危険な がページ上で実行 JavaScript され

ることを制限します。代わりに、回避策として、ライブラリのアセットをページと同じオリジンにホストすることをお勧めしますが推奨されません。

既知の問題と回避策

- iOS モバイルブラウザ (Safari や Chrome など) で録画したコンテンツ (VOD と呼ばれる) を再生する場合、後方シークをするとプレイヤーがミュートになります。

回避策: シーク後に `player.setMuted(false)` を呼び出します。

- iOS モバイルブラウザで録画されたコンテンツを再生する場合、希望の位置を直接選択すると、後方シークが断続的に動作します。

回避方法: シークバーを移動した位置までドラッグします。

- iOS モバイルブラウザで録画したコンテンツを再生するときに、`player.seekTo()` の呼び出しが一貫して機能しません。

回避策: `loadeddata` イベントの後に動画 HTML 要素に `currentTime` を設定します。 例:

```
videoEl.addEventListener('loadeddata', () => {
  videoEl.currentTime = 30; // seek 30s from the beginning
});
```

- iOS モバイル ブラウザでライブ ストリームまたは録画されたコンテンツを再生すると、キャプションが異なるサイズでレンダリングされず、複数回再レンダリングされる場合があります。

回避策: 該当なし。

- iOS モバイルブラウザでライブ配信や録画したコンテンツを再生するときに、`player.getQualities()` 呼び出しで利用可能な品質のリストを返しません。

回避策: 該当なし。プレイヤーは、iOS ブラウザで自動品質モードだけをサポートしています。

- ネイティブ HTML5 コントロールが有効な場合、`setQuality()` の呼び出しは無視されます。

回避策: `player.setQuality()` を呼び出す前に、HTML5 コントロールを無効にします。

- iOS モバイルブラウザでミュート状態のライブストリームを再生すると、非アクティブなプレーヤータブ (タブスイッチやデバイスのロック/ロック解除など) を再開したときに、プレイヤーが不安定 (画面が黒くなる、フリーズする、バッファリングなど) になることがあります。

回避策: JavaScript [ページ可視性 API](#) を使用してページの可視性の変化を検出し、それに応じてプレイヤーに対してアクションを実行します。例:

```
//if client platform is iOS
if (!!navigator.platform && /iPad|iPhone|iPod/.test(navigator.platform)) {
  document.addEventListener("visibilitychange", () => {
    if (document.visibilityState === "hidden" && player.isMuted()) {
      player.pause()
    }
    if (document.visibilityState === "visible" &&
        player.getState() !== PlayerState.PLAYING) {
      player.play()
    }
  })
}
```

Amazon IVS Player SDK: Android ガイド

Amazon Interactive Video Player (IVS) の Android Player SDK は、Android 上で Amazon IVS プレイヤーを使用するために必要なインターフェイスを提供します。

Android モバイルデバイス (スマートフォンとタブレット) でのみ再生パフォーマンスが保証されます。Android TV、Fire TV、IoT デバイス、エミュレーターには対応していません。

com.amazonaws.ivs.player パッケージは、このドキュメントで説明されているインターフェイスを実装します。以下のオペレーションがサポートされています。

- プレーヤーをセットアップ (初期化) します。
- 再生を管理します。
- 品質を管理します。
- イベントを受信します。
- エラーを受信します。

Android プレイヤーの最新バージョン : 1.24.0 ([リリースノート](#))

リファレンスドキュメント : Amazon IVS Android プレイヤーで利用できる最も重要な方法については、<https://aws.github.io/amazon-ivs-player-docs/1.24.0/android/> のリファレンスドキュメントを参照してください。

サンプルコード: GitHub <https://github.com/aws-samples/amazon-ivs-player-android-sample> の Android サンプルリポジトリを参照してください。

プラットフォームの要件: 開発には Android 5.0 (Lollipop) 以上が必要です。

Amazon IVS Player SDK 用の React Native ラッパー が利用可能です。コードとドキュメントについては、<https://github.com/aws/amazon-ivs-react-native-player> を参照してください。

開始方法

ライブラリのインストール

Amazon IVS Android プレイヤーライブラリを Android 開発環境に追加するには、ライブラリを以下に示すモジュールの build.gradle ファイル (最新バージョンの Amazon IVS Player 用) に追加します。

```
repositories {
    mavenCentral()
}

dependencies {
    implementation 'com.amazonaws:ivs-player:1.24.0'
}
```

または、SDK を手動でインストールするには、次の場所から最新バージョンをダウンロードします。

<https://search.maven.org/artifact/com.amazonaws/ivs-player>

プレイヤーの作成とイベントリスナーのセットアップ

プレイヤーのインターフェイスは `com.amazonaws.ivs.player.Player` です。以下に示すように、インターフェイスを初期化します。

```
// Create a player instance
// <this> refers to the current Android Activity
player = Player.Factory.create(this);

// Set up to receive playback events and errors
player.addListener(this);
```

または、`PlayerView` を使用して初期化します。

```
// Create a player instance
// <this> refers to the current Android Activity
PlayerView playerView = new PlayerView(this);
Player player = playerView.getPlayer();
// Set up to receive playback events and errors
player.addListener(this);
```

注: Android アプリケーションのメインスレッドで、リスナーコールバックメソッドが実行されます。

動画の SurfaceView を設定する。

PlayerView を使用しない場合、SurfaceView を Android UI レイアウトに追加して、動画を表示できます。この Surface は、Video Streams 再生前に利用可能である必要があります。基本サーフェスへは、getHolder() の呼び出しによって取得される SurfaceHolder インターフェイスからアクセスできます。(「Android デベロッパリファレンス[SurfaceView](#)」の「」を参照してください)。を使用して SurfaceHolder.Callback、表面の変更に関するイベントを受信します([SurfaceHolder「コールバック」を参照](#))。

```
surfaceView = (SurfaceView) findViewById(R.id.surfaceView);
surfaceView.getHolder().addCallback(this);

@Override
public void surfaceCreated(SurfaceHolder holder) {
    this.surface = holder.getSurface();
    if (player != null) {
        player.setSurface(this.surface);
    }
}

@Override
public void surfaceDestroyed(SurfaceHolder holder) {
    this.surface = null;
    if (player != null) {
        player.setSurface(null);
    }
}
```

ストリームを再生する

ストリームは非同期で読み込まれるため、アプリケーションが `play` メソッドを呼び出して再生を開始する前に、プレイヤーが `READY` 状態である必要があります。 `Player.Listener` インターフェイスを使用して、プレイヤーがいつ正しい状態であるかを判断します。

次のサンプルコードを参照してください。

```
player.load(Uri.parse(url));

@Override
public void onStateChanged(Player.State state) {
    switch (state) {
        case BUFFERING:
            // player is buffering
            break;
        case READY:
            player.play();
            break;
        case IDLE:
            break;
        case PLAYING:
            // playback started
            break;
    }
}
```

プレイヤーを解放する

`player.release()` メソッドは、プレイヤーが使用されなくなったときに呼び出され、ライブラリによって使用されるリソースを解放する必要があります。通常、これはプレイヤーを含むアクティビティまたはフラグメントの `onDestroy` コールバックで行われます。

```
@Override
protected void onDestroy() {
    super.onDestroy();
    player.removeListener(this);
    player.release();
}
```

`player.release()` メソッドの呼び出し後、プレイヤーは使用できなくなります。

アクセス許可

Android Player SDK には、次のアクセス許可が必要です。

```
<uses-permission android:name="android.permission.INTERNET" />
```

さらに、これらのオプションのアクセス許可により、再生体験を向上させることができます。

```
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />  
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
```

スレッドセーフ

プレイヤー API はスレッドセーフではありません。プレイヤーインスタンスへのすべての呼び出しは、同じスレッドからのものである必要があります。

SDK サイズ

Amazon IVS Player SDK は、可能な限りサイズが軽量になるように設計されています。SDK サイズに関する最新情報については、[リリースノート](#)を参照してください。

重要: サイズへの影響を評価する場合、Android Studio で生成した AAB/APK のサイズは、ユーザーのデバイスにダウンロードしたアプリのサイズとは異なりますのでご注意ください。Google Play ストアでは、アプリのサイズを小さくするための最適化が行われています。を使用することをお勧めします。[Android App Bundle](#) を使用して、デバイス設定ごとに最適化されたアプリを提供します。

既知の問題と回避策

- Android プレイヤー SDK は、OkHttp バージョン 4.x にランタイム依存関係があります。OkHttp バージョン 3.x を使用すると、API 署名の不一致や OkHttp 下位互換性の問題により、不安定またはクラッシュする可能性があります。具体的には、プレイヤーは OkHttp バージョン 4.2.2 に依存しますが、任意の 4.x バージョンと互換性がある必要があります。

回避策：4.x バージョンのを使用する OkHttp か、アプリケーション OkHttp からを削除します。

- Android 11 (API レベル 30) エミュレーターを使用すると、動画レイアウトの問題 (特にストリームの拡大表示) が発生することがあります。

回避先: 代わりに、実際のデバイスで再生します。

Amazon IVS Player SDK: iOS ガイド

Amazon Interactive Video Service (IVS) iOS プレイヤーは、iOS で Amazon IVS プレイヤーを使用するために必要なインターフェイスを提供します。

iOS プレイヤーの最新バージョン : 1.24.0 ([リリースノート](#))

リファレンスドキュメント : Amazon IVS iOS プレイヤーで利用できる最も重要な方法については、<https://aws.github.io/amazon-ivs-player-docs/1.24.0/ios/> のリファレンスドキュメントを参照してください。

サンプルコード : : GitHub<https://github.com/aws-samples/amazon-ivs-player-ios-sample> の iOS サンプルリポジトリを参照してください。

プラットフォームの要件: 開発には Xcode 11 以上が必要です。SDK は、物理デバイスと iOS シミュレータの両方で、iOS 12 以降のデプロイターゲットをサポートしています。

Amazon IVS Player SDK 用の React Native ラッパー が利用可能です。コードとドキュメントについては、「<https://github.com/aws/amazon-ivs-react-native-player>」を参照してください。

開始方法

を介してプレイヤー SDK を統合することをお勧めします CocoaPods。(あるいは、フレームワークを手動でプロジェクトに追加することも可能です)。

推奨: Player SDK を統合する (CocoaPods)

リリースは、 という名前 CocoaPods の で公開されます AmazonIVSPlayer。この依存関係を自分の Podfile に追加します。

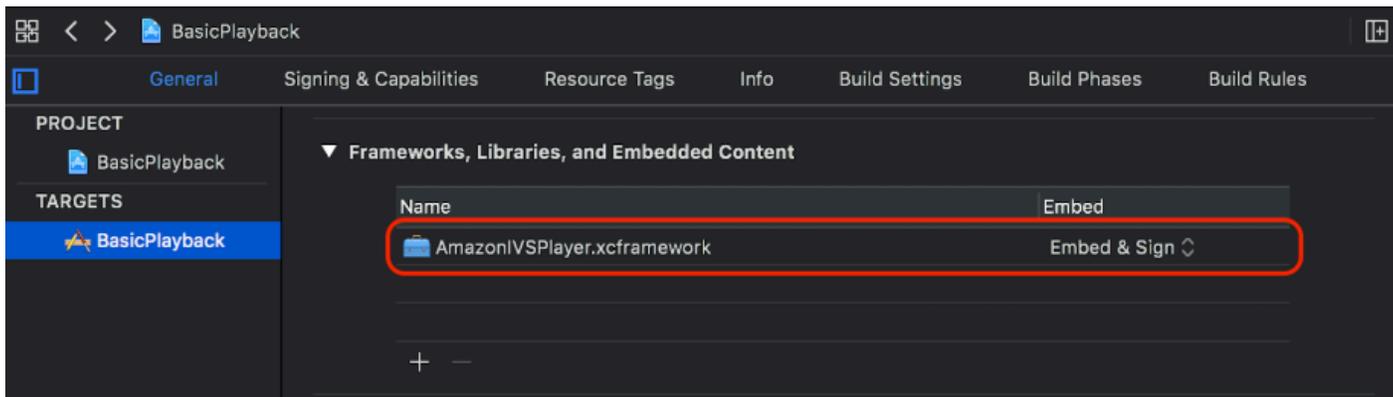
```
pod 'AmazonIVSPlayer'
```

pod install を実行すると、SDK が .xcworkspace で利用できるようになります。

代替方法 : フレームワークを手動でインストールする

1. <https://player.live-video.net/1.24.0/AmazonIVSPlayer.xcframework.zip> から最新バージョンをダウンロードします。
2. アーカイブの内容を抽出します。AmazonIVSPlayer.xcframework には、デバイスとシミュレータの両方の SDK が含まれています。

- アプリケーションターゲットの [全般] タブの、[Frameworks, Libraries, and Embedded Content (フレームワーク、ライブラリ、埋め込みコンテンツ)] のセクションに AmazonIVSPlayer.xcframework をドラッグして埋め込みます。



プレイヤーを作成する

プレイヤーオブジェクトは `IVSPlayer` です。以下のように初期化します。

Swift

```
import AmazonIVSPlayer

let player = IVSPlayer()
```

Objective-C

```
#import <AmazonIVSPlayer/AmazonIVSPlayer.h>

IVSPlayer *player = [[IVSPlayer alloc] init];
```

委任をセットアップする

委任コールバックは、再生状態、イベント、エラーに関する情報を提供します。コールバックはすべて、メインキューで呼び出されます。

Swift

```
// Self must conform to IVSPlayer.Delegate
player.delegate = self
```

Objective-C

```
// Self must conform to IVSPlayer.Delegate
player.delegate = self
```

動画を表示する

プレイヤーは、動画をカスタムレイヤー、IVSPlayerLayer に表示します。SDK は、このレイヤーでバックアップされた UIView サブクラス、IVSPlayerView も提供します。自分のアプリケーションの UI により適したものを使用します。

いずれの場合も、player プロパティを使用して、プレイヤーインスタンスから動画を表示します。

Swift

```
// When using IVSPlayerView:
playerView.player = player

// When using IVSPlayerLayer:
playerLayer.player = player
```

Objective-C

```
// When using IVSPlayerView:
playerView.player = player;

// When using IVSPlayerLayer:
playerLayer.player = player;
```

ストリームをロードする

プレイヤーは、ストリームを非同期的にロードします。再生の準備が整うと、ステータスに表示されます。

Swift

```
player.load(url)
```

Objective-C

```
[player load:url];
```

ストリームを再生する

プレイヤーの準備ができたら、`play` を使って再生を開始します。state プロパティで委任インターフェイスまたはキー値のモニタリングを使用し、状態の変化をモニタリングします。以下は、委任ベースによるアプローチの一例です。

Swift

```
func player(_ player: IVSPlayer, didChangeState state: IVSPlayer.State) {  
    if state == .ready {  
        player.play()  
    }  
}
```

Objective-C

```
-(void)player:(IVSPlayer *)player didChangeState:(IVSPlayerState)state {  
    if (state == IVSPlayerStateReady) {  
        [player play];  
    }  
}
```

アプリのバックグラウンドで一時停止する

プレイヤーは、アプリがバックグラウンドにある間は再生をサポートしませんが、完全に引き離す必要はありません。一時停止すれば十分です。以下の例をご覧ください。

Swift

```
override func viewDidLoad() {  
    super.viewDidLoad()  
  
    NotificationCenter.default.addObserver(self,  
        selector: #selector(applicationDidEnterBackground(_:)),  
        name: UIApplication.didEnterBackgroundNotification,
```

```
        object: nil)
    }

    @objc func applicationDidEnterBackground(_ notification: NSNotification) {
        playerView?.player?.pause()
    }
}
```

Objective-C

```
- (void)viewDidLoad {
    [super viewDidLoad];

    NotificationCenter *defaultCenter = NotificationCenter.defaultCenter;
    [defaultCenter addObserver:self
                     selector:@selector(applicationDidEnterBackground:)
                     name:UIApplicationDidEnterBackgroundNotification
                     object:nil];
}

- (void)applicationDidEnterBackground:(NSNotification *)notification {
    [playerView.player pause];
}
```

スレッドセーフ

プレイヤー API はスレッドセーフではありません。プレイヤーインスタンスは、アプリケーションのメインスレッドから作成して使用する必要があります。

まとめ

以下の単純なビューコントローラーのスニペットは、プレイヤー画面に URL をロードし、再生します。playerView プロパティは、XIB/ストーリーボードから初期化され、そのクラスは [Identity Inspector のカスタムクラスのセクションを使用して](#) インターフェイスビルダーの IVSPlayerView にセットされることに注意してください。

Swift

```
import AmazonIVSPlayer

class MyViewController: UIViewController {
    ...
}
```

```

// Connected in Interface Builder
@IBOutlet var playerView: IVSPlayerView!

override func viewDidLoad() {
    super.viewDidLoad()

    NotificationCenter.default.addObserver(self,
        selector: #selector(applicationDidEnterBackground(_:)),
        name: UIApplication.didEnterBackgroundNotification,
        object: nil)
}

@objc func applicationDidEnterBackground(_ notification: NSNotification) {
    playerView?.player?.pause()
}
...
// Assumes this view controller is already loaded.
// For example, this could be called by a button tap.
func playVideo(url videoURL: URL) {
    let player = IVSPlayer()
    player.delegate = self
    playerView.player = player
    player.load(videoURL)
}
}

extension MyViewController: IVSPlayer.Delegate {
    func player(_ player: IVSPlayer, didChangeState state: IVSPlayer.State) {
        if state == .ready {
            player.play()
        }
    }
}
}

```

Objective-C

```

// MyViewController.h

@class IVSPlayerView;

@interface MyViewController: UIViewController
...
// Connected in Interface Builder

```

```
@property (nonatomic) IBOutlet IVSPlayerView *playerView;
...
@end

// MyViewController.m

#import <AmazonIVSPlayer/AmazonIVSPlayer.h>

@implementation MyViewController <IVSPlayerDelegate>
...

- (void)viewDidLoad {
    [super viewDidLoad];

    NotificationCenter *defaultCenter = NotificationCenter.defaultCenter;
    [defaultCenter addObserver:self
                        selector:@selector(applicationDidEnterBackground:)
                        name:UIApplicationDidEnterBackgroundNotification
                        object:nil];
}

- (void)applicationDidEnterBackground:(NSNotification *)notification {
    [playerView.player pause];
}

// Assumes this view controller is already loaded.
// For example, this could be called by a button tap.
- (void)playVideoWithURL:(NSURL *)videoURL {
    IVSPlayer *player = [[IVSPlayer alloc] init];
    player.delegate = self;
    playerView.player = player;
    [player load:videoURL];
}

- (void)player:(IVSPlayer *)player didChangeState:(IVSPlayerState)state {
    if (state == IVSPlayerStateReady) {
        [player play];
    }
}

...
@end
```

SDK サイズ

Amazon IVS Player SDK は、可能な限りサイズが軽量になるように設計されています。SDK サイズに関する最新情報については、[リリースノート](#)を参照してください。

重要:サイズの影響を評価する場合、Xcode によって生成される IPA のサイズは、ユーザーのデバイスにダウンロードされるアプリのサイズを表すものではありません。App Store では、アプリケーションのサイズを小さくするために最適化が行われています。

既知の問題と回避策

- arm64e アーキテクチャに対するテスト時に、プレイヤーがクラッシュすることがあります。これは、arm64e をターゲットとした場合にのみ該当し、App Store のビルドには該当しません。

回避策: arm64e を使用しないでください。

Amazon IVS Player SDK: Video.js 統合

このドキュメントは、Amazon Interactive Video Service (IVS) Video.js プレイヤーで利用できる最も重要な機能について説明しています。

Video.js プレイヤー統合の最新バージョン: 1.24.0 ([リリースノート](#))

開始方法

Amazon IVS の Video.js のサポートは、Video.js [tech](#) を通じて実装されています。サポートは、スクリプトタグと npm モジュールを介して提供されています。Amazon IVS は、Video.js バージョン 7.6.6 とそれ以降の 7* および 8* をサポートしています。

Video.js の [ソースオプション](#) はサポートされていないので、プレイヤーをインスタンス化するときはご注意ください。代わりに、プレイヤーを通常どおりにインスタンス化し、Video.js の `src()` 関数を呼び出します。自動再生が有効になっていると、ストリームの再生が開始されます。開始されない場合は、`play()` を使って再生を開始します。

デモ

以下のライブデモでは、[Amazon IVS Player Video.js 統合](#) を、当社のコンテンツ配信ネットワークのスクリプトタグと一緒に使用する方法を紹介しています。

スクリプトタグを使ってセットアップする

script タグを使って Amazon IVS tech をセットアップするには

1. 次のタグを含めます (プレイヤー統合の最新バージョン向け)。

```
<script src="https://player.live-video.net/1.24.0/amazon-ivs-videojs-tech.min.js"></script>
```

2. registerIVSTech 関数を使って tech を登録します。

```
registerIVSTech(videojs);
```

videojs は、Video.js から提供されたオブジェクトです。

3. プレイヤーのインスタンスを作成するときは、AmazonIVS を最初の tech として techOrder オプションに追加します。

プレイヤーをインスタンス化するとき、Video.js の [ソースオプション](#) はサポートされません。代わりに、ソースを設定し、プレイヤーを通常通りインスタンス化して、Video.js の src() 関数を呼び出します。自動再生が有効になっていると、ストリームの再生が開始されます。開始されない場合は、play() を使って再生を開始します。

サンプルコード

以下の例では、PLAYBACK_URL がロードするソースストリームです。この例では、Amazon IVS Player の最新バージョンを使用しています。

```
<!doctype html>
<html lang="en">
<head>
  <link href="https://cdnjs.cloudflare.com/ajax/libs/video.js/7.14.3/video-js.css"
    rel="stylesheet">
  <script src="https://cdnjs.cloudflare.com/ajax/libs/video.js/7.14.3/
    video.min.js"></script>
  <script src="https://player.live-video.net/1.24.0/amazon-ivs-videojs-
    tech.min.js"></script>
</head>

<body>
  <div class="video-container">
```

```
<video id="amazon-ivs-videojs" class="video-js vjs-4-3 vjs-big-play-centered"
controls autoplay playsinline></video>
</div>
<style>
  body {
    margin: 0;
  }

  .video-container {
    width: 640px;
    height: 480px;
    margin: 15px;
  }
</style>
<script>
  (function play() {
    // Get playback URL from Amazon IVS API
    var PLAYBACK_URL = '';

    // Register Amazon IVS as playback technology for Video.js
    registerIVSTech(videojs);

    // Initialize player
    var player = videojs('amazon-ivs-videojs', {
      techOrder: ["AmazonIVS"]
    }, () => {
      console.log('Player is ready to use!');
      // Play stream
      player.src(PLAYBACK_URL);
    });
  })();
</script>
</body>
</html>
```

NPM でセットアップする

Amazon IVS プレイヤーを npm から使用するには

1. [video.js](#) npm パッケージをインストールするか、自分のプロジェクトに Video.js ライブラリへの他のアクセス方法があることを確認します。
2. `amazon-ivs-player` npm パッケージをインストールします。

```
npm install amazon-ivs-player
```

3. Amazon IVS tech を登録する準備ができたなら、registerIVSTech 関数をインポートします。

```
import { registerIVSTech } from 'amazon-ivs-player';
```

4. registerIVSTech 関数を使って tech を登録します。

```
registerIVSTech(videojs, options);
```

各パラメータの意味は次のとおりです。

- videojs が、Video.js から提供されたオブジェクトである。
- options が、Amazon IVS tech レイヤーのオプションである。サポートされているオプションは次のとおりです。
 - wasmWorker: amazon-ivs-wasmworker.min.js ファイルがホストされている URL。
 - wasmBinary: amazon-ivs-wasmworker.min.wasm ファイルがホストされている URL。

Worker ファイルは、amazon-ivs-player/dist/ の下の node_modules/ フォルダにあります。IVS プレイヤーを使用するときは、これらをホストする必要があります。

5. プレイヤーのインスタンスを作成するときは、AmazonIVS を最初の tech として techOrder オプションに追加します。

```
const player = videojs('videojs-player', {  
  techOrder: ["AmazonIVS"]  
});
```

TypeScript

を使用している場合 TypeScript、npm パッケージには、インポートして使用できる次のタイプが含まれています。

- VideoJSEvents、getIVSEvents() から返された構造を記述している。
- VideoJSIVSTech、AmazonIVS tech を使用しているプレイヤーインスタンスのインターフェイスを記述している。これは、[@types/video.js](#) npm パッケージが公開している VideoJsPlayer タイプと部分的に重なっている場合があります。

- TechOptions、registerIVSTech() に送信できる設定オプションを定義するインターフェイスを記述している。

上記タイプの詳細については、「[Amazon IVS Player SDK: Web Reference](#)」を参照してください。

イベント

標準の Video.js イベントをリッスンするには、Video.js プレイヤーの `on` 関数を使用します。

Amazon IVS に固有のイベントをリッスンするには、Amazon IVS Web プレイヤーでイベントリスナーを追加および削除します。

```
player.getIVSPlayer().addEventListener(event, callback);
player.getIVSPlayer().removeEventListener(event, callback);
```

この場合、callback はユーザーが定義するコールバックで、event は、PlayerEventType および PlayerState のいずれかです。イベントの詳細については、「[Amazon IVS Player SDK: Web Reference](#)」を参照してください。

エラー

Video.js の一般的なエラーについては、プレイヤー上で汎用の error イベントをリッスンします。

```
player.on("error", callback);
```

Amazon IVS 固有のエラーについては、Amazon IVS プレイヤーでその固有のエラーをリッスンします。

```
let playerEvent = player.getIVSEvents().PlayerEventType;
player.getIVSPlayer().addEventListener(playerEvent.ERROR, callback);
```

コールバックは、次のフィールドを持つオブジェクトを受け取ります。

フィールド	説明
type	エラータイプ。ErrorType イベントに対応しています。詳細については、「 Amazon IVS Player SDK: Web Reference 」を参照してください。

フィールド	説明
code	エラーコードです。
source	エラーのソース。
message	人が判読できるエラーメッセージ。

プラグイン

当社は、利用可能な品質のための UI のトグルを作成するプラグインを提供しています。このプラグインを使用するには、次の `script` タグを通じて当社の tech 使用している場合は、`amazon-ivs-quality-plugin.min.js` ファイルを含めてロードする必要があります (IVS プレイヤーの最新バージョンの場合)。

```
<script src="https://player.live-video.net/1.24.0/amazon-ivs-quality-plugin.min.js"></script>
```

npm を使用している場合は、`amazon-ivs-player` モジュールから `registerIVSQualityPlugin` をインポートします。

```
import { registerIVSQualityPlugin } from 'amazon-ivs-player';
```

次に、Video.js プレイヤーのインスタンスを作成したら、これを登録して有効にするために次の呼び出しを行います。

```
registerIVSQualityPlugin(videojs); // where videojs is the video.js variable  
player.enableIVSQualityPlugin(); // where player is the instance of the videojs player
```

これで、ストリームの品質を選択できる UI メニューのボタンが作成されます。

プラグインと TypeScript

を使用している場合 TypeScript、npm パッケージには、プラグインでインポートして使用できる `VideoJSQualityPlugin` タイプが含まれています。プラグインは実質的に `mixin` であるため、このタイプのインターフェイスは、`VideoJSIVSTech` TypeScript インターフェイスでは [intersection タイプ](#) として使用されます。

コンテンツセキュリティポリシー

Amazon IVS Video.js API は、コンテンツセキュリティポリシー (CSP) を使用しているページで動作するように設定されています。[Amazon IVS Player SDK: Web のガイド](#) の「コンテンツセキュリティポリシーの操作」のセクションを参照してください。

関数

再生

Amazon IVS Video.js API は、Video.js フレームワークによる内部使用に必要なインターフェイスをサポートしています。クライアントアプリケーションでは、おそらく、これらのメソッドを直接使用する必要はありません。Video.js が必要な統合を行い、標準のインターフェイスを表示するためです。ただし、必要に応じて内部の Video.js および Amazon IVS プレイヤーメソッドにアクセスするときは、Video.js プレイヤーオブジェクトを使用して必要なオブジェクトハンドルを tech に付与するのが 1 つの方法です。

API にアクセスするには、通常どおり Video.js プレイヤーのインスタンスを取得します。

```
let player = videojs("videoTagId"); //replace videoTagId with your <video> tag's id
```

次に、そのインスタンスで関数を呼び出します。

以下は、Amazon IVS tech レイヤーがオーバーライドする Video.js 関数のサブセットです。Video.js 関数の完全なリストは、[video.js API documentation](#) を参照してください。

関数	説明および Amazon IVS 固有の情報
currentTime	時間を取得または設定します (開始から秒単位で)。 Amazon IVS: ライブストリームの現在時刻を設定することは推奨されていません。
dispose	プレイヤーインスタンスを削除する。 Amazon IVS: これにより Amazon IVS tech のバックエンドも削除されます。
duration	動画の再生時間を秒単位で返す。

関数	説明および Amazon IVS 固有の情報
	Amazon IVS: ライブストリームの場合、Infinity を返しません。
load	src() データのロードを開始します。 Amazon IVS: これはノーオペレーションです。
play	src コール経由でセットアップされたストリームを再生します。 Amazon IVS: ライブストリームを一時停止した場合、これにより、一時停止した場所からではなく最新のフレームからライブストリームが再開されます。
playbackRate	動画の再生レートを取得または設定します。1.0 は標準速度、0.5 は標準の半分、2.0 は標準の 2 倍、など。 Amazon IVS: ライブストリームでは、1 回の取得で 1 が返され、設定は無視されます。
seekable	シークが可能なメディアの TimeRanges を返します。 Amazon IVS: ライブストリームの場合、戻り値 (TimeRange) で end(0) を呼び出すと、Infinity が返されます。

Amazon IVS に固有の関数

Amazon IVS Video.js tech には、Amazon IVS の機能に固有の動作にアクセスするための追加の関数があります。

関数	説明
getIVSPlayer	基になる Amazon IVS プレイヤーインスタンスを返します。完全な Amazon IVS Player Web API は、このインスタンスから利用できます。可能な限りベーシックな Video.js 再生 API を使用し、Amazon IVS に固有の機能にアクセスするときはこの関数のみを使用することが推奨されます。Amazon IVS プレイ

関数	説明
	ヤーインスタンスでアクセスが必要となる可能性が高い、最も一般的な関数は <code>setQuality()</code> と <code>addEventListener()</code> / <code>removeEventListener()</code> です。
getIVSEvents	Amazon IVS 固有の enum を保持しているオブジェクトを返します。これは、Amazon IVS に固有のエラーをリッスンするために使用されます。詳細については、 イベント および エラー を参照してください。

currentTime

時間を取得または設定します (開始から秒単位で)。

Amazon IVS: ライブストリームの現在時刻を設定することは推奨されていません。

署名

```
currentTime
currentTime(time)
```

パラメータ

パラメータ	Type	説明
time	数値	time がない場合、現在時刻を取得します。time がある場合、動画の再生をその時刻に設定します。

戻り値

Type	説明
number	現在時刻。開始から秒単位で表示します。

dispose

プレイヤーインスタンスを削除します。

Amazon IVS: これにより Amazon IVS tech のバックエンドも削除されます。

署名

```
dispose()
```

パラメータ

なし

戻り値

なし

duration

動画の再生時間を秒単位で返す。

Amazon IVS: ライブストリームの場合、Infinity を返します。

署名

```
duration()
```

パラメータ

なし

戻り値

Type	説明
number	ストリームの継続時間 (秒単位)。ライブストリームの場合、この値は Infinity です。

getIVSEvents

Amazon IVS 固有の enum を保持しているオブジェクトを返します。これは、Amazon IVS 固有のエラーとイベントをリッスンするために使用されます。詳細については、以下を参照してください。

- このドキュメント内の [イベント](#) および [エラー](#)。
- イベント、エラータイプ、およびエラーソースの詳細については、「[Amazon IVS Player SDK: Web Reference](#)」を参照してください。

署名

```
getIVSEvents()
```

パラメータ

なし

戻り値

Type	説明
object	PlayerEventType、PlayerState、ErrorType キーを持つオブジェクト。これらは関連する enum にマップされます。

getIVSPlayer

基になる Amazon IVS プレイヤーインスタンスを返します。完全な Amazon IVS Player Web API は、このインスタンスから利用できます。可能な限りベーシックな Video.js 再生 API を使用し、Amazon IVS に固有の機能にアクセスするときはこの関数のみを使用することが推奨されます。Amazon IVS プレイヤーインスタンスでアクセスが必要となる可能性が高い、最も一般的な関数は、`setQuality()` と `addEventListener()/removeEventListener()` です。

署名

```
getIVSPlayer()
```

パラメータ

なし

戻り値

Type	説明
MediaPlayer	プレイヤーの作成済みインスタンス。

load

`src()` データのロードを開始します。

Amazon IVS: これはノーオペレーションです。

署名

```
load()
```

パラメータ

なし

戻り値

なし

play

`src` コール経由でセットアップされたストリームを再生します。

Amazon IVS: ライブストリームを一時停止した場合、これにより、一時停止した場所からではなく最新のフレームからライブストリームが再開されます。

署名

```
play()
```

パラメータ

なし

戻り値

なし

playbackRate

動画の再生レートを取得または設定します。1.0 は標準速度、0.5 は標準の半分、2.0 は標準の 2 倍、など。

Amazon IVS: ライブストリームでは、1 回の取得で 1 が返され、設定は無視されます。

署名

```
playbackRate
playbackRate(rate)
```

パラメータ

パラメータ	Type	説明
rate	number	再生レート。有効な値は [0.25, 2.0] の範囲です。

戻り値

Type	説明
number	再生レート。

seekable

シークが可能なメディアの TimeRanges を返します。

Amazon IVS: ライブストリームの場合、戻り値 (TimeRange) で end(0) を呼び出すと、Infinity が返されます。

署名

```
seekable()
```

パラメータ

なし

戻り値

Type	説明
TimeRange	TimeRange シークに使用できるメディアの。

Amazon IVS Player SDK: JW プレイヤー統合

このドキュメントでは、Amazon Interactive Video Service (IVS) JW Player 統合で利用可能な最も重要な機能について説明します。

JW Player 統合の最新バージョン : 1.24.0 ([リリースノート](#))

開始方法

JW Player に対する Amazon IVS サポートは、プロバイダーを通じて実装されます。Amazon IVS プロバイダーは、JW Player の Web プレイヤーでのみサポートされています。プロバイダーはスクリプトタグを介してロードされます。Amazon IVS プロバイダーの再生を必要とするストリームは、プレイリストの type: 'ivs' でタグ付けされている必要があります。Amazon IVS は JW Player のバージョン 8.18.4 以降をサポートしています。

セットアップ

以下の手順において、JW_PLAYER_DIV は JW Player インスタンスの <div> の名前です。IVS_STREAM は、IVS 再生 URL です。Amazon IVS プロバイダーをセットアップして再生を有効にするには

1. 次の script タグを含めます (プレイヤー統合の最新バージョンの場合は、この場合は 1.24.0)。

```
<script src="https://player.live-video.net/1.24.0/amazon-ivs-jw-provider.min.js"></script>
```

2. `ivs`タイプを使用して、IVS プレイリスト項目をマークします。`setup()`の`cast`値を`null`へ設定し、`Chromecast` はサポートされていないため。

```
jwplayer(JW_PLAYER_DIV).setup({
  playlist: [{
    file: IVS_STREAM,
    type: 'ivs',
  }]
});
```

3. 基盤となる Amazon IVS Player への参照を使用して Amazon IVS Player API を呼び出す場合や、コールバック処理のために Amazon IVS 固有の列挙型への参照が必要な場合は、`'providerPlayer'` イベントヘリスナーを追加します。

```
jwplayer(JW_PLAYER_DIV).on('providerPlayer', function (player) {
  // player object has 'ivsPlayer' and 'ivsEvents' properties
  // ...callback code...
});
```

サンプルコード

この例では、`JW_PLAYER_LIB`は JW Player ライブラリスクリプトへの URL で、`IVS_STREAM`は IVS 再生 URL です。

```
<!DOCTYPE html>
<html lang="en">
<head>
  <script src=JW_PLAYER_LIB></script>
  <script src="https://player.live-video.net/1.24.0/amazon-ivs-jw-provider.min.js"></script>
</head>
<body>
  <div id='player'></div>
  <script>
    // set default values for ivsPlayer and ivsEvents
    var ivsPlayer = {};
    var ivsEvents = {};

    // define our player setup
    const ivsConfig = {
      playlist: [{
```

```
        file: IVS_STREAM,
        type: 'ivs',
    ]}
};

jwplayer('player')
    .setup(ivsConfig)
    .on('providerPlayer', function (player) {
        console.log('Amazon IVS Player: ', player.ivsPlayer);
        console.log('Amazon IVS Player Events: ', player.ivsEvents);

        // store the reference to the Amazon IVS Player
        ivsPlayer = player.ivsPlayer;
        // store the reference to the Amazon IVS Player Events
        ivsEvents = player.ivsEvents;
    });
</script>
</body>
</html>
```

イベント

標準の JW Player イベントをリッスンするには、JW Player の[オン](#)関数を使用します。

Amazon IVS 固有のイベントをリッスンする、または Amazon IVS Web プレイヤーでイベントリスナーを追加または削除するには、'providerPlayer' イベントをリッスンして Amazon IVS Player への参照を取得し、それにイベントリスニングを追加します。例:

```
// store a default value for ivsPlayer
var ivsPlayer = {};

// store references to the Amazon IVS Player and Amazon IVS Events:
jwplayer(JW_PLAYER_DIV).on('providerPlayer', function (player) {
    ivsPlayer = player.ivsPlayer;
});

// set up event listening
ivsPlayer.addEventListener(event, callback);
ivsPlayer.removeEventListener(event, callback);
```

callback はユーザーの定義するコールバックで、event は PlayerEventType、PlayerState、ErrorTypeのいずれかです。イベントの詳細については、「[Amazon IVS Player SDK: Web Reference](#)」を参照してください。

'providerPlayer' イベントは JW Player によって発行され、そのイベントで登録したコールバックは、次のフィールドを持つオブジェクトを受け取ります。

フィールド	説明
ivsPlayer	基になる Amazon IVS プレイヤーインスタンスを返します。完全な Amazon IVS Player Web API は、このインスタンスから利用できます。ベーシックな JW Player 再生 API を可能な限り使用し、この機能を使用して Amazon IVS 固有の機能へアクセスすることをお勧めします。Amazon IVS プレイヤーインスタンスでアクセスする必要がある最も一般的な機能は、addEventListener() およびremoveEventListener() です。
ivsEvents	関連付けられた Amazon IVS 固有の列挙型にマップされた、PlayerEventType、PlayerState、ErrorType フィールドを含むオブジェクトを返します。詳細については、「 Amazon IVS Player SDK: Web Reference 」を参照してください。

エラー

一般的な JW Player のエラーについては、JW Player の[オン](#)関数を使用して、エラーイベントをリスンしてください。

Amazon IVS 固有のエラーについては、Amazon IVS プレイヤーでその固有のエラーをリスンします。

```
// set default values for ivsPlayer and ivsEvents
var ivsPlayer = {};
var ivsEvents = {};

// store references to the Amazon IVS Player and Amazon IVS Events
jwplayer(JW_PLAYER_DIV).on('providerPlayer', function (player) {
  ivsPlayer = player.ivsPlayer;
  ivsEvents = player.ivsEvents;
});
```

```
// set up event listening:  
let playerEvent = ivsEvents.PlayerEventType;  
ivsPlayer.addEventListener(playerEvent.ERROR, callback);
```

コールバックは、次のフィールドを持つオブジェクトを受け取ります。

フィールド	説明
type	エラータイプ。ErrorType イベントに対応しています。詳細については、「 Amazon IVS Player SDK: Web Reference 」を参照してください。
code	エラーコードです。
source	エラーのソース。
message	人が判読できるエラーメッセージ。

コンテンツセキュリティポリシー

Amazon IVS プロバイダー API は、コンテンツセキュリティポリシー (CSP: Content Security Policy) を使用するページで動作するように設定されています。[Amazon IVS Player SDK: Web のガイド](#)の「コンテンツセキュリティポリシーの操作」のセクションを参照してください。

制限事項

プロバイダーはキャストをサポートしていません。JW Player ダッシュボードでキャストを有効にした場合は、`setup()` を呼び出すときに `cast ~ null` を設定することで無効にできます。これにより、キャストボタンが非表示になります。

動画ストリーム内にメタデータを埋め込む

Amazon Interactive Video Service (IVS) の時間指定メタデータは、Amazon IVS ストリームにメタデータを埋め込む方法を提供します。これにより、ストリームのレイテンシーや地理的位置にかかわらず、すべての視聴者が同時に動画ストリームでメタデータを受信できるようになります。

時間指定メタデータとは

時間指定メタデータは、タイムスタンプ付きのメタデータです。IVS API または IVS broadcast SDK を使用して、プログラムでストリームに挿入できます。Amazon IVS がストリームを処理するとき、時間指定メタデータはオーディオフレームおよび動画フレームと同期されます。再生中、ストリームのすべての視聴者は、ストリームに対してメタデータを同時に取得します。タイムコードはキューポイントとして機能します。キューポイントを使用すると、次のようなデータに基づいてアクションをトリガーできます。

- スポーツストリームの選手統計を更新する。
- ライブショッピングストリームの製品詳細を送信する。
- ライブクイズストリームの質問を送信する。

Amazon IVS の時間指定メタデータは、動画セグメントに埋め込まれた ID3 タグを使用します。その結果、録画された動画でメタデータが利用可能になります。

IAM アクセス許可のセットアップ

前提条件: [IAM ユーザーの作成とアクセス許可の設定](#) (IAM ユーザーの作成とアクセス許可の設定を含む) の手順を実行しておく必要があります。

次に、時間指定メタデータを使用するアクセス許可を IAM ユーザーに付与する必要があります。以下のステップに従ってください。

1. AWS マネジメントコンソールにサインインして、IAM コンソールを開きます。 <https://console.aws.amazon.com/iam/>
2. ナビゲーションペインで、[Users (ユーザー)] を選択します。次に目的のユーザー (AWS アカウントの作成時に指定したユーザー名) を選択します。
3. ユーザーの [Summary (概要)] ウィンドウの [Permissions (アクセス許可)] タブで [Add inline policy (インラインポリシーの追加)] (右側) を選択します。

4. [JSON] タブで、この blob に貼り付けます。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ivs:PutMetadata"
      ],
      "Resource": "arn:aws:ivs:*:*:channel/*"
    }
  ]
}
```

- 引き続き [Create Policy (ポリシーの作成)] ウィンドウで、[Review Policy (ポリシーの確認)] を選択します。ポリシーの [Name (名前)] を入力し、[Create policy (ポリシーの作成)] を選択します。
- ユーザーの [Summary (概要)] ウィンドウが開き、新しいポリシー名が表示されます。

時間指定メタデータの挿入

時間指定メタデータは、指定されたチャンネルのアクティブなストリームにのみ挿入できます。

AWS CLI の使用

テストで、時間指定メタデータを追加する最も簡単な方法は、AWS CLI を使用する方法です。AWS CLI を使用するには、まず、マシンで CLI をダウンロードして設定する必要があります。[IVS の開始方法](#) を開始する を実行した際に CLI を取得していなかった場合は、今すぐ取得してください。詳細については、「[AWS Command Line Interface のユーザーガイド](#)」を参照してください。

CLI の取得後

- put-metadata コマンドを実行し、チャンネル ARN とメタデータを渡します。

```
aws ivs put-metadata --channel-arn <your-channel-arn> --metadata <your-metadata>
```

例:

```
aws ivs put-metadata --channel-arn arn:aws:ivs:us-west-2:465369119046:channel/GbiYJna5hFoC --metadata '{"question": "What does IVS stand for?", "correctIndex":
```

```
0, "answers": ["interactive video service", "interesting video service", "ingenious video service"]}]'
```

- Amazon IVS は、ストリームがライブかどうかをチェックします。ストリームがライブでない場合はエラーが発生します。ライブの場合は、CLI がエラーなしで戻り、メタデータ (テキスト blob) がストリームに挿入されます。これはすぐに実行されます。実行のタイミングは保証されませんが、ストリーム内の同じポイントですべての視聴者にメタデータが表示されます。

Amazon IVS API の使用

プログラムで時間指定メタデータを挿入するには、[PutMetadata](#) エンドポイントを使用します。

HTTP リクエストの例を次に示します。

```
POST /PutMetadata HTTP/1.1
{
  "channelArn": "my_channel",
  "metadata": "{\"question\": \"What does IVS stand for?\", \"correctIndex\": 0, \"answers\": [\"interactive video service\", \"interesting video service\", \"ingenious video service\"]}"
}
```

IVS Broadcast SDK の使用

IVS Broadcast SDK を使用して時間指定メタデータを帯域内に挿入できます。これは、メタデータをオーディオおよびビデオコンテンツと同期させるのに便利な場合があります。

- Android — BroadcastSession クラスでは、sendTimedMetadata を使用してください。
- iOS — IVSBroadcastSession クラスでは、sendTimedMetadata を使用してください。

時間指定メタデータの使用

Amazon IVS Player を使用して、動画ストリームに埋め込まれた時間指定メタデータを使用します。[Amazon IVS Player SDK](#) およびその他のプレイヤードキュメントを参照してください。

以下は、Amazon IVS Player SDK を使用してコンソールに受信したメタデータを印刷する例です。再生が、メタデータが埋め込まれたセグメントに到達すると、イベントがトリガーされます。(イベントは、ウェブ用が TEXT_METADATA_CUE、Android 用が onCue()、iOS 用が player(_:didOutputCue:) です)。このイベントを使用して、インタラクティブウィジェットの

更新など、クライアントアプリケーション内の機能を開始できます。このイベントは、ライブコンテンツと録画コンテンツの両方に対してトリガーされます。

Amazon IVS Player SDK for Web:

```
const player = IVSPlayer.create();
player.addEventListener(IVSPlayer.PlayerEventType.TEXT_METADATA_CUE,
  function (cue) {
    console.log('Timed metadata: ', cue.text);
  });
```

Amazon IVS Player SDK for Android:

```
@Override
public void onCue(@NonNull Cue cue) {
    if(cue instanceof TextMetadataCue) {
        Log.i("Timed Metadata: ", ((TextMetadataCue)cue).text);
    }
}
```

Amazon IVS Player SDK for iOS:

```
func player(_ player: IVSPlayer, didOutputCue cue: IVSCue) {
    if let textMetadataCue = cue as? IVSTextMetadataCue {
        print("Timed Metadata: \(textMetadataCue.text)")
    }
}
```

注: 時間指定メタデータは、Player 1.3.0 以降の iOS Safari および iOS Chrome でサポートされています。

サンプルデモ: クイズアプリ

インタラクティブなクイズアプリのコード例は、GitHub で入手できます。時間指定メタデータを介して JSON を使用し、クイズ UI を設定して質問と回答を表示します。回答は選択可能であり、選択が正しいかどうかが表示されます。

Amazon IVS Player SDK プラットフォーム	サンプルのリポジトリ
Web	https://github.com/aws-samples/amazon-ivs-basic-web-sample

Amazon IVS Player SDK プラットフォーム	サンプルのリポジトリ このリポジトリ内で、 クイズデモ (および ライブデモ) を参照してください。
Android	https://github.com/aws-samples/amazon-ivs-player-android-sample このリポジトリ内で、 クイズデモ を参照してください。
iOS	https://github.com/aws-samples/amazon-ivs-player-ios-sample このリポジトリ内で、 クイズデモ を参照してください。

時間指定メタデータの表示

必要に応じて、ライブストリームに埋め込まれた時間指定メタデータをコンソールで表示できます。

1. [Amazon IVS コンソール](#)を開きます。
2. 左上のハンバーガーアイコンを選択してナビゲーションペインを開き、[Live channels (ライブチャンネル)] を選択します。
3. ストリームを表示するチャンネルを選択し、そのチャンネルの詳細ページに移動します。

ライブストリームは、詳細ページのライブストリームセクションで再生されています。

4. ウィンドウの下部にある [Timed Metadata (時間指定メタデータ)] を選択します。

プレイヤーの再生中に各時間指定メタデータイベントを受信すると、その値と受信時間が表示されます。

追加情報

Amazon IVS の時間指定メタデータの使用に関する 2 部構成のブログシリーズの最初の部分である、[Using Amazon Interactive Video Service Timed Metadata](#) を参照してください。

プライベートチャンネルの設定

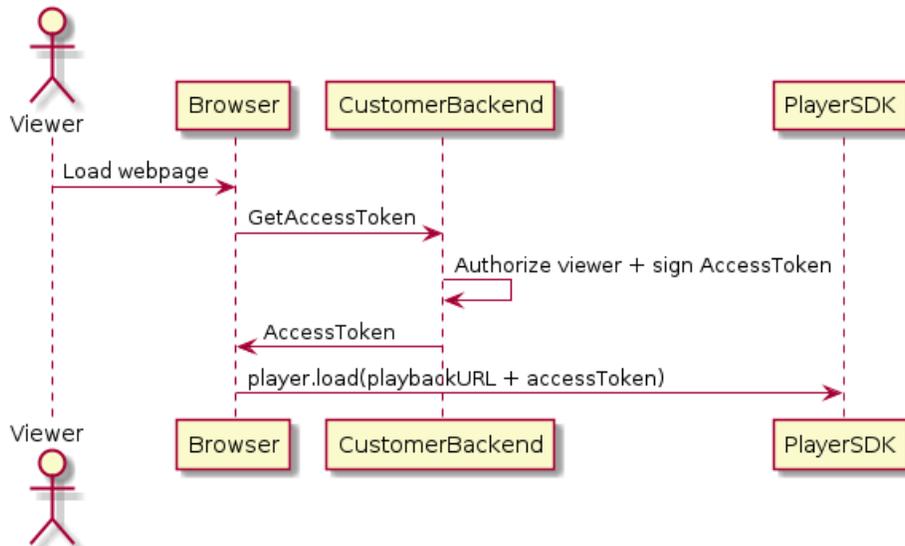
Amazon Interactive Video Service (IVS) では、プライベートチャンネルを作成できるため、チャンネルや視聴者ごとにストリームを制限できます。チャンネルの再生承認を有効にして、承認された再生リクエスト用の署名付き JSON Web Token (JWT) を生成することにより、動画再生へのアクセスを管理します。

チャンネルでの再生承認リクエストはオプションです。視聴者がストリームを視聴しようとするときに、チャンネルで承認が有効になっている場合、Amazon IVS は視聴者がリクエストに有効な再生トークンを持っていることを確認します。再生トークンは、Amazon IVS のユーザーが (再生承認キーを使用して) 署名する JWT であり、再生承認が有効になっているチャンネルのすべての再生リクエストを含んでいます。

トピック

- [プライベートチャンネルのワークフロー](#)
- [再生キーの作成またはインポート](#)
- [チャンネルでの再生承認を有効にする](#)
- [再生トークンの生成と署名](#)
- [再生キーのリスト化](#)
- [再生キーの削除](#)
- [再生キー情報の取得](#)
- [ビューアーセッションを取り消す](#)

プライベートチャンネルのワークフロー



1. 視聴者がプライベートストリームのウェブページを読み込もうとすると、ブラウザはアクセストークンをリクエストします。(ユーザーはこれを行うためのブラウザコードを提供します)。
2. ユーザーのバックエンドアプリはアクセストークンリクエストを受け取り、その視聴者にストリームを表示する権限を付与するかを判断します。権限を付与する場合、バックエンドは JWT を生成し、ユーザーのプライベートキーを使用して署名し、署名された JWT を再生リクエストでブラウザに返します。
3. ブラウザは、Amazon IVS プレーヤー (または別のプレーヤー) SDK へのリクエストを使用して、ストリームをロードします。リクエストには、ストリーム再生 URL と署名された JWT が含まれます。
4. Amazon IVS は、ユーザーのパブリックキーを使用して、JWT が正しいプライベートキーを使用して署名されたことを確認します。
5. JWT が確認されると、Amazon IVS は視聴者のプライベートストリームを再生します。

ユーザーは、以下を作成する責任があります。

- アクセストークンをリクエストするブラウザコード。
- JWT を生成して署名するバックエンドサーバーアプリ。
- 再生承認キーペア。これには、AWS が保持するパブリックキーとユーザーがダウンロードするプライベートキーの 2 つのキーがあります。プライベートキーを使用して、プライベートチャンネルへのアクセスを許可する JWT に署名します。

上記の方法 (ブラウザからのネットワークリクエストを使用してトークンをフェッチする) 以外に、再生承認を実装する方法があります。代わりに、最初のウェブページで署名された再生トークンを送信して、視聴者が行う必要のあるネットワークラウンドトリップの回数を減らすことができます。

以下のセクションでは、チャンネルをプライベートにする (再生承認を有効にする)、再生トークンを生成、署名する、再生キーペアを操作する方法について説明します。

注: 以下のコンソール手順で、左ナビゲーションメニューが表示されない場合は、左上にあるハンバーガーアイコンを選択して開くことができます。

再生キーの作成またはインポート

Amazon IVS では、再生トークンの署名と検証に使用できるキーペアを最大 3 つまで使用できます。Amazon IVS はキーローテーションを提供していません。

一度インポートされた後は、再生キーを更新できません。代わりに、既存の再生キーを削除して新しいキーを読み込む必要があります。

[ECDSA パブリック/プライベートキーペア](#)を生成して JWT に署名し、パブリックキーを再生キーリソースとして Amazon IVS にアップロードします。その後、Amazon IVS は再生リクエストで署名を検証できます。

キーペアの作成

キーペアを作成するにはさまざまな方法があります。以下に 2 つの例を示します。

コンソールを使用して新しいキーペアを作成するには、次の手順に従います。このプロセスでは、プライベートキーのみをダウンロードできることに注意してください。

1. [Amazon IVS コンソール](#)を開きます。チャンネルのリージョンを選択していない場合は、リージョンを選択します。
2. 左側のナビゲーションメニューで、セキュリティの再生 > キーの再生 を選択します。
3. [Create playback key (再生キーの作成)] を選択します。[Create playback key (再生キーの作成)] ダイアログが表示されます。
4. プロンプトに従って、[Create (作成)] を選択します。
5. Amazon IVS が新しいキーペアを生成します。パブリックキーは再生キーリソースとしてインポートされ、プライベートキーはすぐにダウンロードできるようになります。

Amazon IVS はクライアント側でキーを生成し、プライベートキーを保存しません。キーは必ず保存してください。後で取得することはできません。

OpenSSL で P384 EC キーペアを作成するには (最初に [OpenSSL](#) のインストールが必要な場合があります)、次の手順を行います。このプロセスにより、プライベートキーとパブリックキーの両方にアクセスできます。パブリックキーは、トークンの検証をテストする場合にのみ必要です。

```
openssl ecparam -name secp384r1 -genkey -noout -out priv.pem
openssl ec -in priv.pem -pubout -out public.pem
```

次に、以下の手順に従い、新しいパブリックキーをインポートします。

既存のパブリックキーをインポートするには

既存のキーペアがある場合は、パブリックキーを IVS にインポートできます。プライベートキーはシステムでは必要ありませんが、ユーザーがトークンを署名するのに使用されます。

コンソールで既存のパブリックキーをインポートするには

1. [Amazon IVS コンソール](#)を開きます。チャンネルのリージョンを選択していない場合は、リージョンを選択します。
2. 左側のナビゲーションメニューで、**セキュリティの再生 > キーの再生** を選択します。
3. [Import (インポート)] を選択します。[Import playback key (再生キーのインポート)] ダイアログが表示されます。
4. プロンプトに従って、[Import (インポート)] を選択します。
5. Amazon IVS はパブリックキーをインポートし、再生キーリソースを生成します。

CLI で既存のパブリックキーをインポートするには、次のコマンドを実行します。

```
aws ivs import-playback-key-pair --public-key-material "`cat public.pem`" --region
<aws-region>
```

リージョンがローカルの AWS 設定ファイルにある場合、`--region <aws-region>` を省略できます。

レスポンスの例を次に示します。

```
{
  "keyPair": {
    "arn": "arn:aws:ivs:us-west-2:693991300569:playback-key/f99cde61-
c2b0-4df3-8941-ca7d38acca1a",
    "fingerprint": "98:0d:1a:a0:19:96:1e:ea:0a:0a:2c:9a:42:19:2b:e7",
    "tags": {}
  }
}
```

API リクエスト

```
POST /ImportPlaybackKeyPair HTTP/1.1
{
  "publicKeyMaterial": "<pem file contents>"
}
```

チャンネルでの再生承認を有効にする

チャンネルの承認要件は、チャンネルの作成時または作成後に (更新エンドポイントを使用して) 設定できます。再生承認を有効または無効にする手順は同じです。

コンソールの手順

チャンネルの作成時に承認を有効にするには

1. [Amazon IVS コンソール](#)を開きます。チャンネルのリージョンを選択していない場合は、リージョンを選択します。
2. [Get started (開始する)] ボックス (右上) で [Create channel (チャンネルの作成)] を選択します。
3. [Create channel (チャンネルの作成)] ページで、[Custom configuration (カスタム設定)] を選択します。
4. [Playback authentication (再生承認)] セクションで、[Enable token-authentication requirement for video playback (動画再生のトークン認証要件を有効にする)] を有効にします。
5. 残りのプロンプトに従って、チャンネルを作成します。(「[IVS の開始方法](#)」を参照してください。)

既存のチャンネルを更新して承認を有効にするには

1. [Amazon IVS コンソール](#)を開きます。チャンネルのリージョンを選択していない場合は、リージョンを選択します。

2. 左のナビゲーションメニューの [Channels (チャンネル)] を選択します。
3. 更新するチャンネルのチェックボックスを選択し、[Edit (編集)] を選択します。
4. [Playback authentication (再生承認)] セクションで、[Enable token-authentication requirement for video playback (動画再生のトークン認証要件を有効にする)] を有効にします。
5. [Save changes (変更の保存)] をクリックします。

CLI の手順

チャンネルの作成時に承認を有効にするには

```
aws ivs create-channel --authorized --region <aws-region>
```

リージョンがローカルの AWS 設定ファイルにある場合、`--region <aws-region>` を省略できます。

レスポンスの例を次に示します。authorized は true に設定されています。

```
{
  "streamKey": {
    "channelArn": "arn:aws:ivs:us-west-2:123456789:channel/fbc789c1-2c56-4ce6-a30a-d99275dc4481",
    "value": "sk_us-west-2_abcd1234efgh5678ijkl",
    "arn": "arn:aws:ivs:us-west-2:123456789:stream-key/62f15f1b-fe31-4127-b252-0666ac7f55a7",
    "tags": {}
  },
  "channel": {
    "name": "test-channel",
    "tags": {},
    "authorized": true,
    "latencyMode": "LOW",
    "ingestEndpoint": "jds34ksdg3las.global-contribute.live-video.net",
    "playbackUrl": "https://b37c565f6d79.us-west-2.playback.live-video.net/api/video/v1/aws.ivs.us-west-2.123456789.channel.oU40KS4LA1Dz.m3u8",
    "arn": "arn:aws:ivs:us-west-2:123456789:channel/fbc789c1-2c56-4ce6-a30a-d99275dc4481"
  }
}
```

既存のチャンネルを更新して承認を有効にするには

```
aws ivs update-channel --arn
arn:aws:ivs:us-west-2:693991300569:channel/742da049-fe9f-4f23-928e-c6753760a189
--authorized
```

これは一例です。--arn の後に独自のチャンネル ARN を指定する必要があります。チャンネルを作成する場合と同様に、更新レスポンスの authorized は true です。

API リクエスト (作成と更新)

```
POST /CreateChannel HTTP/1.1
{
  "name": "<your channel name>",
  "authorized": true
}
```

```
POST /UpdateChannel HTTP/1.1
{
  "arn": "<channel arn>",
  "authorized": true
}
```

再生トークンの生成と署名

JWT およびトークン署名用にサポートされているライブラリの操作の詳細については、jwt.io を参照してください。jwt.io インターフェイスでは、プライベートキーを入力してトークンに署名する必要があります。パブリックキーは、トークンを検証する場合にのみ必要です。

トークンスキーマ

すべての JWT には、ヘッダー、ペイロード、署名の 3 つのフィールドがあります。

- ヘッダーは以下を指定します。
 - alg は署名アルゴリズムです。これは、SHA-384 ハッシュアルゴリズムを使用する ECDSA 署名アルゴリズムの ES384 です。
 - typ はトークン型、JWT です。

```
{
  "alg": "ES384",
```

```
"typ": "JWT"  
}
```

- ペイロードには、Amazon IVS 固有のデータが含まれています。
- `channel-arn` は、動画再生リクエストのリファレンスです。
- `access-control-allow-origin` は、再生を指定された[オリジン](#)に制限するために使用できるオプションのフィールドです。つまり、指定されたウェブサイトからのみストリームを表示できるようにします。たとえば、誰かが他のウェブサイトにプレイヤーを埋め込むのを防ぐことができます。デフォルトでは、すべてのオリジンで再生が許可されます。(これにより、ブラウザクライアントのみが制限され、ブラウザ以外のクライアントからの再生は制限されません)。このフィールドには、カンマで区切られた複数のオリジンを含めることができます。ワイルドカードドメインが許可されており、各オリジンのホスト名を * で始めることができます (例: `https://*.amazon.com`)。
- `strict-origin-enforcement` は、`access-control-allow-origin` フィールドで指定されたオリジンの制限を強化するために使用できるオプションのフィールドです。デフォルトでは、`access-control-allow-origin` 制限はマルチバリエーションプレイリストにのみ適用されます。`strict-origin-enforcement` が有効になっている場合、サーバーは、リクエストを実行しているオリジンがすべての再生リクエスト (マルチバリエーションプレイリスト、バリエーションプレイリスト、およびセグメントを含む) のトークンと一致するという要件を強制します。つまり、すべてのクライアント (ブラウザ以外のクライアントを含む) は、各リクエストで有効なオリジンリクエストヘッダーを提供する必要があります。setOrigin メソッドを使用して、IVS iOS および Android プレイヤー SDK でヘッダーを設定します。iOS Safari を除くウェブブラウザでは自動的に設定されます。iOS Safari の場合、オリジンリクエストヘッダーが確実に送信されるように、動画要素に `crossorigin="anonymous"` を追加する必要があります。例えば、`<video crossorigin="anonymous"></video>` などです。
- `single-use-uuid` は、[トークンの作成時に生成する有効な汎用識別子 \(UUID\)](#) を含むオプションのフィールドです。このフィールドと UUID 値を追加すると、生成した関連トークンは、マルチバリエーションプレイリストの取得やストリームの視聴に使用された時点で無効になります。一回限りの認証トークンを使用すると、悪意のあるユーザーにとって、プライベートチャンネルのストリームを他の視聴者と共有することが難しくなります。`single-use-uuid` クレームを使用する場合、`exp` クレームの最大値は 10 分後になることに注意してください。
- `viewer-id` は、トークンが付与された視聴者を追跡したり参照したりするのに使用される ID を含むオプションのフィールドです。このフィールドは、今後視聴者の視聴セッションの取り消しを有効にするために必要です。最大長は 40 文字で、値は文字列である必要があります。このフィールドを個人を特定する情報、機密情報、または機密情報には使用しないでください。`viewer-id` を使用する場合、`exp` の最大値は 10 分後になることに注意してください。

- `viewer-session-version` は、このビューアーセッションに関連付けるバージョンを含むオプションフィールドです。ビューアーセッションを取り消す場合、この値を使用してどのビューアーセッションを取り消すかをフィルタリングできます。たとえば、ここで Unix タイムスタンプを指定すると、指定した時間より前に開始されたすべてのセッションを取り消すことができます。値は 64 ビットの符号付き整数 (Int64) でなければなりません。このフィールドは、`viewer-id` と一緒に (オプションで) 提供されることを意図しており、それ自体は何もしません。デフォルト値は 0 です。
- `exp` は、トークンの有効期限を示す Unix UTC タイムスタンプです。これは、ストリームを表示できる時間の長さを示すものではありません。トークンは、ストリーム全体を通してではなく、視聴者が再生を初期化するときに検証されます。この値を整数で入力します。

Unix タイムスタンプは、うるう秒を無視した、1970-01-01T00:00:00Z UTC から指定された UTC 日付/時刻までの秒数を表す数値であることに注意してください。言語が異なると、Unix タイムスタンプは異なる単位で測定されます。例えば、JavaScriptは時間をミリ秒単位で `Date.now()` 返します。(「[JWT RFC section 4.1.4](#)」の「`exp`」を参照してください。)

```
{
  "aws:channel-arn": "<channel_arn>",
  "aws:access-control-allow-origin": "<your-origin>",
  "aws:strict-origin-enforcement": true,
  "aws:single-use-uuid": "<UUID>",
  "aws:viewer-id": "<viewer_id>",
  "aws:viewer-session-version": "<viewer_session_version>",
  "exp": <unix timestamp>
}
```

- 署名を作成するには、ヘッダー (ES384) で指定されたアルゴリズムを使用して、エンコードされたヘッダーとエンコードされたペイロードに署名します。

```
ECDSASHA384(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  <private-key>
)
```

手順

1. ES384 署名アルゴリズムと、再生キーリソース (上記の ECDSASHA384 例を参照) の 1 つに関連付けられているプライベートキーを使用してトークンの署名を生成します。
2. トークンを組み立てます。

```
base64UrlEncode(header) + "." +  
base64UrlEncode(payload) + "." +  
base64UrlEncode(signature)
```

3. 署名付きトークンをクエリパラメータとして、再生 URL に追加します。

```
https://b37c565f6d790a14a0e78afaa6808a80.us-west-2.playback.live-video.net/  
api/video/v1/aws.ivs.us-west-2.123456789.  
channel.fbc789c1-2c56-4ce6-a30a-d99275dc4481.m3u8?token=<token>
```

再生キーのリスト化

Amazon IVS では、いつでもすべての再生キーリソースのリストを取得できます。

コンソールの手順

1. [Amazon IVS コンソール](#)を開きます。チャンネルのリージョンを選択していない場合は、リージョンを選択します。
2. 左側のナビゲーションメニューで、**セキュリティの再生 > キーの再生**を選択します。

アカウントに関連付けられているすべての再生キーリソースが表示されます。削除されたキーは表示されず、過去のキーの履歴も表示されません。

CLI の手順

```
aws ivs list-playback-key-pairs --region <aws-region>
```

リージョンがローカルの AWS 設定ファイルにある場合、`--region <aws-region>` を省略できます。

レスポンスの例:

```
{
  "keyPairs": [
    {
      "arn": "arn:aws:ivs:us-west-2:991729659840:playback-key/3db9fc15-df57-4c02-b5a6-d4ee3448b8ad",
      "fingerprint": "81:f3:8c:88:78:61:4e:bc:58:07:a3:ca:63:f5:72:08",
      "tags": {}
    },
    {
      "arn": "arn:aws:ivs:us-west-2:991729659840:playback-key/3ff88c71-b18e-415f-948b-18bbde605a97",
      "fingerprint": "a2:b5:b3:0b:be:8e:73:00:0e:ad:e9:bb:02:c9:81:9a",
      "tags": {}
    }
  ]
}
```

API リクエスト

詳細については、「IVS 低レイテンシーストリーミング API リファレンス [ListPlaybackKeyPairs](#)」の「」を参照してください。

```
POST /ListPlaybackKeyPairs HTTP/1.1
{
  "maxResults": number,
  "nextToken": "string"
}
```

再生キーの削除

Amazon IVS では、アカウントから再生キーを削除できます。キーを削除すると、アカウントからリソースが削除されます。削除されたキーで署名された再生トークンは、検証に合格しません。

コンソールの手順

1. [Amazon IVS コンソール](#)を開きます。チャンネルのリージョンを選択していない場合は、リージョンを選択します。
2. 左側のナビゲーションメニューで、セキュリティの再生 > キーの再生 を選択します。
3. 削除するキーを選択します。

- [Delete (削除)] を選択します。[Delete playback (再生の削除)] キーダイアログが表示されます。
- [Delete playback key (再生キーの削除)] を選択します。

CLI の手順

キーの ARN がある場合は、AWS CLI を使用して再生キーを削除できます。Amazon IVS は、CLI による Batch 削除をサポートしていません。

```
aws ivs delete-playback-key-pair --arn arn:aws:ivs:us-west-2:991729659840:playback-key/3db9fc15-df57-4c02-b5a6-d4ee3448b8ad --region <aws-region>
```

リージョンがローカルの AWS 設定ファイルにある場合、`--region <aws-region>` を省略できます。

成功した場合はレスポンスがありません。以下の `get` コマンドを実行して、キーが削除されたことを確認します。

エラーレスポンスの例を次に示します。

```
An error occurred (ResourceNotFoundException) when calling the DeletePlaybackKeyPair operation: ResourceNotFoundException:
```

API リクエスト

```
POST /DeletePlaybackKeyPair HTTP/1.1
{
  "arn": "<playback key arn>"
}
```

再生キー情報の取得

Amazon IVS では、再生キーリソースに関する情報を取得できます。Amazon IVS のコンソール経由で再生キーが作成された場合でも、関連するプライベートキーは使用できないことに注意してください。

コンソールの手順

1. [Amazon IVS コンソール](#)を開きます。チャンネルのリージョンを選択していない場合は、リージョンを選択します。
2. 左側のナビゲーションメニューで、セキュリティの再生 > キーの再生 を選択します。
3. 詳細を取得するキーを選択し、[View details (詳細を表示)] を選択します。

CLI の手順

```
aws ivs get-playback-key-pair --arn arn:aws:ivs:us-west-2:991729659840:playback-key/3db9fc15-df57-4c02-b5a6-d4ee3448b8ad --region <aws-region>
```

リージョンがローカルの AWS 設定ファイルにある場合、`--region <aws-region>` を省略できます。

レスポンスの例:

```
{
  "keyPair": {
    "arn": "arn:aws:ivs:us-west-2:991729659840:playback-key/3ff88c71-b18e-415f-948b-18bbde605a97",
    "fingerprint": "a2:b5:b3:0b:be:8e:73:00:0e:ad:e9:bb:02:c9:81:9a",
    "tags": {}
  }
}
```

API リクエスト

```
POST /GetPlaybackKeyPair HTTP/1.1
{
  "arn": "<playback key arn>"
}
```

ビューアーセッションを取り消す

Amazon IVS の顧客は、認証トークンに関連付けられたビューアーセッションを取り消して、そのトークンを使用した再生を禁止または停止できます。ユースケースとして、パブリックストリームを

プライベートストリームに移行して、パブリックストリームの一部の視聴者だけが引き続き視聴できるようにすることが挙げられます。

以下の手順に記載されている viewer-id フィールドの情報については、[the section called “再生トークンの生成と署名”](#) の「トークンスキーマ」を参照してください。

CLI の手順

チャンネル ARN とビューアー ID があれば、AWS CLI を使用してビューアーセッションを取り消すことができます。

```
aws ivs start-viewer-session-revocation --channel-arn arn:aws:ivs:us-west-2:991729659840:channel/abcdABCDefgh --viewer-id UDbh1u6M8nr0oarrzuKe --region <aws-region>
```

オプション入力で、`--viewer-session-versions-less-than-or-equal-to <version>` はビューアーセッションのどのバージョンを一度に取り消すかのフィルターを指定できます。

リージョンがローカルの AWS 設定ファイルにある場合、`--region <aws-region>` を省略できます。

成功した場合はレスポンスがありません。

エラーレスポンスの例を次に示します。

```
An error occurred (ValidationException) when calling the StartViewerSessionRevocation operation: ValidationException:
```

API リクエスト

```
POST /StartViewerSessionRevocation HTTP/1.1
{
  "channelArn": <channel ARN>,
  "viewerId": <viewer ID>,
  "viewerSessionVersionsLessThanOrEqualTo": <version>
}
```

BatchStartViewerSessionRevocation エンドポイントもあります。[IVS 低レイテンシーストリーミング API リファレンス](#)を参照してください。

Amazon S3 への自動録画 (Low-Latency Streaming)

このセクションでは、Amazon IVS 低レイテンシーストリーミングの auto-record-to-S3 機能について説明します。録画された Amazon IVS ストリームのデータストレージについて説明します。ストレージの内容とメタデータファイルスキーマについて説明します。また、録画したコンテンツの再生についても説明します。

詳細	参照
動画録画の設定と停止	Amazon IVS の使用開始で、 オプション録画を使用したチャンネルを作成する
API	IVS API リファレンス
コスト	Amazon IVS のコスト

S3 プレフィックス

S3 プレフィックスは、録画される各ライブストリームの一意的ディレクトリ構造です。ライブストリームのすべてのメディアファイルとメタデータファイルは、このディレクトリ内に書き込まれます。録画が有効になっているチャンネルの場合、S3 プレフィックスはライブセッションの開始時に生成され、録画の開始と終了時に CloudWatch イベントで提供されます。

S3 プレフィックスの形式は次のとおりです。

```
/ivs/v1/<aws_account_id>/<channel_id>/<year>/<month>/<day>/<hours>/<minutes>/<recording_id>
```

実行する条件は以下のとおりです。

- aws_account_id は、チャンネルが作成される AWS アカウントの (AWS アカウントの作成時に生成される) ID です。
- channel_id は、チャンネル ARN のリソース ID 部分 (Amazon リソースネームの最後の部分) です。[用語集](#) の ARN を参照してください。
- <year>/<month>/<day>/<hours>/<minutes> は、録画開始時の UTC タイムスタンプです。
- recording_id は、録画セッションごとに生成される一意の ID です。

例:

```
ivs/v1/123456789012/AsXego4U6tnj/2020/6/23/20/12/j8Z9091ndcVs
```

録画の内容

録画が開始すると、動画セグメントとメタデータファイルは、チャンネルに設定されている S3 バケツトに書き込まれます。これらのコンテンツは、後処理またはオンデマンド動画再生として利用できません。

ライブストリームが開始され、Recording Start EventBridge イベントが出力されると、マニフェストファイルとビデオセグメントが書き込まれるまでに少し時間がかかることに注意してください。Recording End イベントの送信後に、録画したストリームを再生または処理することをお勧めします。(「[IVS EventBridge での Amazon の使用](#)」を参照してください。)

以下は、ライブの Amazon IVS セッションの録画のディレクトリ構造およびコンテンツの例です。

```
ivs/v1/123456789012/AsXego4U6tnj/2020/6/23/20/12/j8Z9091ndcVs/  
  events  
    recording-started.json  
    recording-ended.json  
  media  
    hls  
    thumbnails
```

events フォルダには、録画イベントに対応するメタデータファイルが含まれています。JSON メタデータファイルは、録画が開始された時、正常に終了した時、失敗して終了した時に生成されます。

- events/recording-started.json
- events/recording-ended.json
- events/recording-failed.json

該当する events フォルダには、recording-started.json、および recording-ended.json または recording-failed.json のどちらかが含まれます。

これらには、録画されたセッションとその出力形式に関連するメタデータが含まれます。JSON の詳細を以下に示します。

media フォルダには、サポートされているすべてのメディアコンテンツが 2 つのサブフォルダ内に格納されています。

- hls には、ライブセッション中に生成されたすべてのメディアファイルとマニフェストファイルが含まれており、Amazon IVS プレイヤーで再生できます。このフォルダには、標準マスターマニフェスト master.m3u8 とバイト範囲対応マニフェスト byte-range-multivariant.m3u8 の 2 種類の HLS マニフェストがあります。したがって、各レンディションフォルダには playlist.m3u8 と byte-range-variant.m3u8 ファイルの両方があります。(以下の「[バイト範囲プレイリスト](#)」を参照してください。)
- thumbnails には、ライブセッション中に生成されたサムネイル画像が含まれます。サムネイルは、毎分ごとに生成されてバケットに書き込まれます。(この動作を変更するには、録画設定の thumbnailConfiguration プロパティを上書きしてください。)

重要: media フォルダ内のコンテンツは動的に生成され、最初に受信した動画セグメントの特性によって決定されます。フォルダのコンテンツは最終的な特性 (レンディションの品質など) を表していない場合があります。静的パスについて何も仮定しないでください。使用可能な HLS レンディションとそのパスを検出するには、以下で説明する JSON メタデータファイルを使用します。

バイト範囲プレイリスト

auto-record-to-S3 機能は、標準の HLS [プレイリストに加えて、バイト範囲](#)プレイリスト生成をサポートします。バイト範囲プレイリストは、HLS 仕様のバージョン 4 に準拠しています。これにより、より詳細なコンテンツクリッピングが可能になります。バイト範囲プレイリストでは、レンディションインデックスファイルの各セグメントはビデオチャンクのバイトのサブ範囲を参照し、標準の 10 秒のメディアファイルサイズよりも細かく指定できます。バイト範囲のプレイリストでは、セグメント再生時間はストリームに設定されたキーフレーム間隔と同じです。

サムネイル

録画設定の thumbnailConfiguration プロパティは、ライブセッションのサムネイルの記録を有効または無効にしたり、ライブセッションのサムネイルが生成される間隔を変更したりできます。サムネイルの間隔は 1 秒から 60 秒の範囲で設定でき、デフォルトでは、60 秒の間隔でサムネイルの記録が有効になっています。詳細については、「[Amazon IVS API リファレンス](#)」を参照してください。

また、サムネイル設定には storage フィールド (SEQUENTIAL または LATEST) と解像度 (LOWEST_RESOLUTION、SD、HD、FULL_HD) が含まれています。各オプションの解像度は以下のとおりです。

160 <= LOWEST_RESOLUTION <= 360

360 < SD <= 480

480 < HD <= 720

720 < FULL_HD <= 1,080

断片化されたストリームの結合

録画設定の recordingReconnectWindowSeconds プロパティを使用すると、ストリームが中断されて新しいストリームが開始された場合に、Amazon IVS が前のストリームと同じ S3 プレフィックスに録画を試みる時間枠 (秒単位) を指定できます。つまり、ブロードキャストが切断され、指定された時間内で再接続された場合、複数のストリームは 1 つのブロードキャストと見なされマージされます。

Amazon での IVS 録画状態変更イベント EventBridge : Amazon IVS が新しいストリームが開始されないことを確認するのを待機するため recordingReconnectWindowSeconds、Recording End イベントと recording-ended JSON メタデータファイルは少なくとも遅延します。

ストリームの結合機能をセットアップする手順については、「Amazon IVS の開始方法」の [「ステップ 4: 任意の録画によるチャンネルの作成」](#) を参照してください。

対象

複数のストリームを同じ S3 プレフィックスに録画するには、すべてのストリームで特定の条件を満たす必要があります。

- 動画の幅と高さが同じであること。
- フレームレートは同じであること。
- 後続ストリームのビットレート差は、元のストリームのビットレートの 50% 以下であること。
- ビデオコーデックとオーディオコーデックは同じであること。

注意:

- 最大 20 のストリームがマージされ、その後新しい S3 プレフィックスが作成されます。

- 48 時間後、新しい S3 プレフィックスが作成されます。たとえば、最初のブロードキャストが 48 時間続き、別のブロードキャストが `recordingReconnectWindowSeconds` の指定時間内に開始された場合、次のブロードキャストは最初の S3 プレフィックスにはマージされません。
- 各ストリームの開始は、前のストリームから 10 秒以上経過後に行う必要があります。

既知の問題

もし `recordingReconnectWindowSeconds` が有効になっていて、Web Broadcast SDK が使用されている場合、Web Broadcast SDK はビットレートと品質を動的に変更するため、同じ S3 プレフィックスへの録画が機能しない可能性があります。

JSON メタデータファイル

録画状態変更イベントが発生すると、対応する Amazon CloudWatch メトリクスが生成され、メタデータファイルが S3 プレフィックス内に書き込まれます。(「[Amazon IVS Low-Latency Streaming のモニタリング](#)」を参照してください。)

このメタデータは JSON 形式です。これには、以下の情報が含まれています。

フィールド	Type	必須	説明
<code>channel_arn</code>	文字列	はい	ライブストリームをブロードキャストするチャンネルの ARN。
<code>media</code>	オブジェクト	はい	この録画に使用できるメディアコンテンツの列挙型オブジェクトを含むオブジェクト。有効な値: "hls"、"thumbnails"。
<code>hls</code>	オブジェクト	はい	Apple HLS 形式の出力を記述する列挙型フィールド。
<code>duration_ms</code>	整数	条件付き	録画された HLS コンテンツの継続時間 (ミリ秒単位)。これは、 <code>recording_status</code> が "RECORDING_ENDED" または "RECORDING_ENDED_W

フィールド	Type	必須	説明
			ITH_FAILURE" のときにのみ利用できます。録画完了前に障害が発生した場合は、0 になります。
path	文字列	はい	HLS コンテンツが格納されている S3 プレフィックスからの相対パス。
playlist	文字列	はい	HLS マスタープレイリストファイルの名前。
byte_range_playlist	文字列	はい	HLS バイト範囲マルチバリエーションプレイリストの名前。
renditions	オブジェクト	はい	メタデータオブジェクトのレンディション (HLS バリエーション) の配列。レンディションは必ず 1 つ以上。
path	文字列	はい	このレンディションの HLS コンテンツが格納されている S3 プレフィックスからの相対パス。
playlist	文字列	はい	このレンディションのメディアプレイリストファイルの名前。
byte_range_playlist	文字列	はい	このレンディションのバイト範囲プレイリストの名前。
resolution_height	整数	条件付き	エンコードされた動画のピクセル解像度の高さ。これは、レンディションに動画トラックが含まれている場合にのみ使用できます。

フィールド	Type	必須	説明
resolution_width	整数	条件付き	エンコードされた動画のピクセル解像度の幅。これは、レンダリングに動画トラックが含まれている場合にのみ使用できます。
thumbnails	オブジェクト	条件付き	サムネイル出力を記述する列挙型フィールド。これを使用できるのは、サムネイル設定の recordingMode が INTERVAL になっているときだけです。
path	文字列	条件付き	サムネイルコンテンツが格納されている S3 プレフィックスからの相対パス。これを使用できるのは、サムネイル設定の recordingMode が INTERVAL になっているときだけです。
resolution_height	整数	はい	サムネイルの高さ。デフォルト: ソースレンダリングの解像度。この値は、関連する録画設定でのユーザー入力の影響を受けます。具体的には thumbnailConfiguration.resolution 値です。
resolution_width	整数	はい	サムネイルの幅。デフォルト: ソースレンダリングの解像度。この値は、関連する録画設定でのユーザー入力の影響を受けます。具体的には thumbnailConfiguration.resolution 値です。

フィールド	Type	必須	説明
<code>latest_thumbnail</code>	オブジェクト	はい	最新のサムネイル出力を記述する列挙型フィールド。これを使用できるのは、サムネイル設定の <code>storage</code> に <code>LATEST</code> が含まれる場合のみです。
<code>resolution_height</code>	整数	はい	サムネイルの高さ。デフォルトは、ソースレンディションの解像度です。この値は、関連する録画設定でのユーザー入力の影響を受けます。具体的には <code>thumbnailConfiguration.resolution</code> 値です。
<code>resolution_width</code>	整数	はい	サムネイルの幅。デフォルトは、ソースレンディションの解像度です。この値は、関連する録画設定でのユーザー入力の影響を受けます。具体的には <code>thumbnailConfiguration.resolution</code> 値です。

フィールド	Type	必須	説明
recording_ended_at	string	条件付き	<p>録画終了時の RFC 3339 UTC タイムスタンプ。これは、recording_status が "RECORDING_ENDED" または "RECORDING_ENDED_WITH_FAILURE" のときにのみ利用できます。</p> <p>recording_started_at と recording_ended_at は、これらのイベントが生成されたときのタイムスタンプであり、HLS ビデオセグメントのタイムスタンプと完全に一致しない場合があります。録画時間を正確に決定するには、duration_ms フィールドを使用してください。</p>
recording_started_at	文字列	はい	<p>録画開始時の RFC 3339 UTC タイムスタンプ。</p> <p>recording_ended_at については、上記の注意事項を参照してください。</p>
recording_status	文字列	はい	<p>録画ステータス。有効な値は、"RECORDING_STARTED"、"RECORDING_ENDED"、"RECORDING_ENDED_WITH_FAILURE" です。</p>

フィールド	Type	必須	説明
recording_status_message	文字列	条件付き	ステータスの詳細情報。これは、recording_status が "RECORDING_ENDED" または "RECORDING_ENDED_WITH_FAILURE" のときにのみ利用できます。
version	文字列	はい	メタデータスキーマのバージョン。

例: recording_started.json

```
{
  "version" : "v1",
  "channel_arn" : "arn:aws:ivs:us-west-2:123456789012:channel/AsXego4U6tnj",
  "recording_started_at" : "2020-06-12T12:53:26Z",
  "recording_status" : "RECORDING_STARTED",
  "media" : {
    "hls" : {
      "path" : "media/hls",
      "playlist" : "master.m3u8",
      "byte_range_playlist": "byte-range-multivariant.m3u8",
      "renditions" : [
        {
          "path" : "480p30",
          "playlist" : "playlist.m3u8",
          "byte_range_playlist": "byte-range-variant.m3u8",
          "resolution_height" : 480,
          "resolution_width" : 852
        },
        {
          "path" : "360p30",
          "playlist" : "playlist.m3u8",
          "byte_range_playlist": "byte-range-variant.m3u8",
          "resolution_height" : 360,
          "resolution_width" : 640
        }
      ]
    }
  }
}
```

```
    "path" : "160p30",
    "playlist" : "playlist.m3u8",
    "byte_range_playlist": "byte-range-variant.m3u8",
    "resolution_height" : 160,
    "resolution_width" : 284
  },
  {
    "path" : "720p60",
    "playlist" : "playlist.m3u8",
    "byte_range_playlist": "byte-range-variant.m3u8",
    "resolution_height" : 720,
    "resolution_width" : 1280
  }
]
},
"thumbnails": {
  "path": "media/thumbnails",
  "resolution_height": 480,
  "resolution_width": 852
},
"latest_thumbnail": {
  "path": "media/latest_thumbnail/thumb.jpg",
  "resolution_height": 480,
  "resolution_width": 852
}
}
```

例: recording_ended.json

```
{
  "version" : "v1",
  "channel_arn" : "arn:aws:ivs:us-west-2:123456789012:channel/AsXego4U6tnj",
  "recording_ended_at" : "2020-06-14T12:53:20Z",
  "recording_started_at" : "2020-06-12T12:53:26Z",
  "recording_status" : "RECORDING_ENDED",
  "media" : {
    "hls" : {
      "duration_ms" : 172794489,
      "path" : "media/hls",
      "playlist" : "master.m3u8",
      "byte_range_playlist": "byte-range-multivariant.m3u8",
      "renditions" : [
```

```
{
  {
    "path" : "480p30",
    "playlist" : "playlist.m3u8",
    "byte_range_playlist": "byte-range-variant.m3u8",
    "resolution_height" : 480,
    "resolution_width" : 852
  },
  {
    "path" : "360p30",
    "playlist" : "playlist.m3u8",
    "byte_range_playlist": "byte-range-variant.m3u8",
    "resolution_height" : 360,
    "resolution_width" : 640
  },
  {
    "path" : "160p30",
    "playlist" : "playlist.m3u8",
    "byte_range_playlist": "byte-range-variant.m3u8",
    "resolution_height" : 160,
    "resolution_width" : 284
  },
  {
    "path" : "720p60",
    "playlist" : "playlist.m3u8",
    "byte_range_playlist": "byte-range-variant.m3u8",
    "resolution_height" : 720,
    "resolution_width" : 1280
  }
]
},
"thumbnails": {
  "path": "media/thumbnails",
  "resolution_height": 480,
  "resolution_width": 852
},
"latest_thumbnail": {
  "path": "media/latest_thumbnail/thumb.jpg",
  "resolution_height": 480,
  "resolution_width": 852
}
}
```

例: recording_failed.json

```
{
  "version" : "v1",
  "channel_arn" : "arn:aws:ivs:us-west-2:123456789012:channel/AsXego4U6tnj",
  "recording_ended_at" : "2020-06-14T12:53:20Z",
  "recording_started_at" : "2020-06-12T12:53:26Z",
  "recording_status" : "RECORDING_ENDED",
  "media" : {
    "hls" : {
      "duration_ms" : 172794489,
      "path" : "media/hls",
      "playlist" : "master.m3u8",
      "byte_range_playlist": "byte-range-multivariant.m3u8",
      "renditions" : [
        {
          "path" : "480p30",
          "playlist" : "playlist.m3u8",
          "byte_range_playlist": "byte-range-variant.m3u8",
          "resolution_height" : 480,
          "resolution_width" : 852
        },
        {
          "path" : "360p30",
          "playlist" : "playlist.m3u8",
          "byte_range_playlist": "byte-range-variant.m3u8",
          "resolution_height" : 360,
          "resolution_width" : 640
        },
        {
          "path" : "160p30",
          "playlist" : "playlist.m3u8",
          "byte_range_playlist": "byte-range-variant.m3u8",
          "resolution_height" : 160,
          "resolution_width" : 284
        },
        {
          "path" : "720p60",
          "playlist" : "playlist.m3u8",
          "byte_range_playlist": "byte-range-variant.m3u8",
          "resolution_height" : 720,
          "resolution_width" : 1280
        }
      ]
    }
  }
}
```

```
    ]
  },
  "thumbnails": {
    "path": "media/thumbnails",
    "resolution_height": 480,
    "resolution_width": 852
  },
  "latest_thumbnail": {
    "path": "media/latest_thumbnail/thumb.jpg",
    "resolution_height": 480,
    "resolution_width": 852
  }
}
```

録画のレンディションの検出

Amazon IVS チャンネルにコンテンツをストリーミングすると、auto-record-to-s3 はソースビデオを使用して複数のレンディションを生成します。Amazon IVS Player は、[Adaptive Bitrate Streaming \(ABR\)](#) を使用して、必要に応じてレンディション (ビットレート) を自動的に切り替え、さまざまなネットワーク条件に応じて再生を最適化します。

ライブストリーミング中に生成された各レンディションは、S3 録画プレフィックス内の固有のパスに記録されます。解像度の詳細、パス、およびプレイリストファイル名は、録画の開始および終了時に [JSON メタデータファイル](#) に保存されます。録画設定の renditionSelection の値が ALL の場合、すべてのレンディションが記録対象として選択されます。renditionSelection が CUSTOM の場合、オプション (LOWEST_RESOLUTION、SD、HD、FULL_HD) から 1 つ以上を選択する必要があります。各オプションの解像度は以下のとおりです。

160 <= LOWEST_RESOLUTION <= 360

360 < SD <= 480

480 < HD <= 720

720 < FULL_HD <= 1,080

重要: 静的レンディションのパスまたは生成されたレンディションのリストを仮定しないでください。これらは変更される可能性があります。特定のレンディションが Amazon IVS 録画で常に使用

可能であると前提にしないでください。利用可能なレンディション、解像度、パスを決定するには、メタデータファイルを参照してください。

録画プレフィックス内の event/recording_started.json または event/recording_ended.json ファイルには、録画プレフィックス内のメディアファイルのパスと名前が含まれています。すべての path 要素は、階層内の前のパスに関連しています。media > hls の要素は HLS アセットを表しており、マスタープレイリスト名とパスがこのレベルで定義されています。

S3 録画プレフィックスとメタデータファイルを使用して、マスタープレイリストパスを生成する方法を示す Python コードスニペットを次に示します。

```
def get_master_playlist(metadata_json, s3_recording_prefix):
    return s3_recording_prefix + '/' + metadata_json['media']['hls']['path'] + '/' +
        metadata_json['media']['hls']['playlist']
```

media > hls > renditions の要素は、記録されたレンディションのリストを表します。resolution_height および resolution_width プロパティは、動画の解像度を特定するために使用されます。path および playlist 要素は、レンディションプレイリストパスを取得するために使用されます。これらのフィールドを使用して、後処理に使用するレンディションを決定します。

録画に使用できる最高のレンディションプレイリストを見つけるには、「IVS Recording State Change EventBridge」イベントにサブスクライブできます。(「[IVS EventBridge での Amazon の使用](#)」を参照してください。) これらのイベントへサブスクライブされた Lambda 関数の使用例を示す Python スクリプトを以下に示します。

```
import json
import boto3
s3 = boto3.resource('s3')

def get_highest_rendition_playlist(bucket_name, prefix_name):
    object_path = "{}/events/recording-started.json".format(prefix_name)
    object = s3.Object(bucket_name, object_path)
    body = str(object.get()['Body'].read().decode('utf-8'))
    metadata = json.loads(body)
    media_path = metadata["media"]["hls"]["path"]
    renditions = metadata["media"]["hls"]["renditions"]

    highest_rendition = None
```

```
highest_rendition_size = 0

for rendition in renditions:
    current_rendition_size = rendition["resolution_height"]
    if (current_rendition_size > highest_rendition_size):
        highest_rendition_size = current_rendition_size
        highest_rendition = rendition

highest_rendition_playlist = media_path + '/' + highest_rendition['path'] + '/' +
highest_rendition['playlist']
return highest_rendition_playlist

def lambda_handler(event, context):
    prefix_name = event["detail"]["recording_s3_key_prefix"]
    bucket_name = event["detail"]["recording_s3_bucket_name"]
    rendition_playlist = get_highest_rendition_playlist(bucket_name, prefix_name)
    print("Highest rendition playlist: {}/{}".format(prefix_name, rendition_playlist))

    return {
        'statusCode': 200,
        'body': rendition_playlist
    }
```

プライベートバケットからの録画コンテンツの再生

Amazon S3 への自動録画機能で録画されたオブジェクトは、デフォルトではプライベートです。したがって、これらのオブジェクトは S3 のダイレクト URL を使用してアクセスして再生することはできません。Amazon IVS プレーヤーまたは別のプレーヤーを使用して再生するために HLS マスターマニフェスト (m3u8 ファイル) を開こうとすると、エラー (「You do not have permission to access the requested resource」など) が表示されます。代わりに、これらのファイルを Amazon CloudFront CDN (コンテンツ配信ネットワーク) で再生できます。

Amazon CloudFront デイストリビューション

CloudFront デイストリビューションは、プライベートバケットからコンテンツを配信するように設定できます。通常、これは読み取りが提供するコントロールをバイパスするオープンにアクセス可能なバケットを持つよりも望ましいです CloudFront。オリジンアクセスコントロール (OAC) を作成することで、プライベートバケットからサービスするようにデイストリビューションを設定できます。OAC は、プライベートオリジンバケットに対する読み取り権限を持つ特別な CloudFront ユー

ザーです。OAC は、ディストリビューションを作成した後、CloudFront コンソールまたは API を使用して作成できます。「[新しいオリジンアクセスコントロールの作成](#)」を参照してください。

Amazon からの再生 CloudFront

OAC を使用してディストリビューションを設定してプライベートバケットにアクセスすると、動画ファイルを URL から CloudFront 使用できるようになります。CloudFront URL は、AWS CloudFront コンソールの詳細タブのディストリビューションドメイン名です。次のように表示されます。

```
a1b23cdef4ghij.cloudfront.net
```

録画した動画をディストリビューションでストリーミングするには、`master.m3u8` ファイルのオブジェクトキーを使います。次のように表示されます。

```
ivs/v1/012345678912/a0bCDeFGH1IjK/2021/4/20/12/03/aBcdEFghIjkl/media/hls/master.m3u8
```

オブジェクトキーを CloudFront URL の末尾に追加します。最終的にページ URL は次のようになります。

```
https://a1b23cdef4ghij.cloudfront.net/ivs/v1/012345678912/  
a0bCDeFGH1IjK/2021/4/20/12/03/aBcdEFghIjkl/media/hls/master.m3u8
```

ウェブブラウザから再生するには、必ずと S3 バケットの両方で CloudFront CORS を設定してください。CloudFront 設定については、「[オリジンリクエストポリシーの作成](#)」の手順に従って、CORS-S3 オリジンリクエストポリシーと SimpleCORS レスポンスヘッダーポリシーを CloudFront ディストリビューションにアタッチします。以下の設定コンソールページの例を参照してください。

Cache key and origin requests

We recommend using a cache policy and origin request policy to control the cache key and origin requests.

Cache policy and origin request policy (recommended)

Legacy cache settings

Cache policy

Choose an existing cache policy or create a new one.

CachingOptimized

Recommended for S3

Policy with caching enabled. Supports Gzip and Brotli compression.

[Create cache policy](#) [View policy](#)

Origin request policy - optional

Choose an existing origin request policy or create a new one.

CORS-S3Origin

Policy for S3 origin with CORS

[Create origin request policy](#) [View policy](#)

Response headers policy - optional

Choose an existing response headers policy or create a new one.

SimpleCORS

Allows all origins for simple CORS requests

[Create response headers policy](#) [View policy](#)

▶ Additional settings

S3 CORS 設定については、「[CORS の設定](#)」を参照して、S3 バケットに適切なルールを作成してください。

録画した動画を、バケットから直接再生しているかのように再生できるようになりました。

詳細については、「[Amazon S3 オリジンへのアクセスの制限](#)」を参照してください。

IVS 低レイテンシーストリーミング EventBridge での Amazon の使用

Amazon Interactive Video Service (IVS) ストリーム EventBridge をモニタリングするには、Amazon を使用できます。

Amazon IVS は、ストリームのステータスに関する変更イベントを Amazon に送信します EventBridge。配信されたすべてのイベントが有効です。ただし、イベントはベストエフォートベースで送信されます。つまり、以下を保証するものではありません。

- イベントが配信される — 指定されたイベント (ストリームの開始など) は発生しますが、Amazon IVS が対応する変更イベントを に送信しない可能性があります EventBridge。Amazon IVS は、配信を中止する前に、数時間にわたりイベントの配信を試みます。
- 配信されたイベントが指定された時間内に到着する — 数時間前のイベントを受け取ることもあります。
- イベントが順序通りに配信される — イベントは、特に短い時間内に送信された場合、順不同になることがあります。例えば、Stream Up の前に Stream Down が配信されることもあります。

イベントが欠落したり、遅延したり、順序が違ったりすることはまれですが、通知イベントの順序や存在に依存するビジネスクリティカルなプログラムを作成するときは、こうした可能性に対処しておく必要があります。

次のいずれかのイベントに対して EventBridge ルールを作成できます。

イベントタイプ	イベント	配信するタイミング
IVS Stream State Change	Session Created	チャンネルストリームキーが使用され、ストリームセッションが正常に作成されました。このイベントは、ビデオが処理または視聴者に配信される前に、ストリームが開始されたときに発生します。これは、ストリームが開始されたが、設定ミスや制限違反などの原因で配信に失敗したかどうかを判断するのに役立ちます。

イベントタイプ	イベント	配信するタイミング
IVS Stream State Change	Session Ended	<p>エンコーダが切断され、Amazon IVS はビデオを受信しなくなりました。このイベントは、いつエンコーダがメディアの送信を停止したか特定するのに役立ちます。</p> <p>注意:エンコーダが切断されると、Session Ended イベントが Stream End イベントの前に発生することがあります。これは Amazon IVS が引き続きビデオを処理している場合、Session Ended イベント後に短い時間が発生することがあるためです。</p>
IVS Stream State Change	Stream Start	<p>ストリームが処理され、視聴者が番組を視聴できるようになります。このイベントは、ビデオストリームが処理中であり、視聴者が視聴できることを示します。これは、ストリームが正常に配信されたかどうかを判断するのに役立ちます。</p>
IVS Stream State Change	Stream End	<p>ストリームの処理が停止し、視聴者に動画セグメントが提供されなくなります。このイベントは、いつストリームが終了し、視聴者が新しいビデオセグメントを使用できなくなったかを判断するのに役立ちます。(「セッション終了」の注記も参照してください)。</p>
IVS Stream State Change	Stream Failure	<p>処理能力を超えたため、ストリームは処理されず、利用できなくなります。</p>

イベントタイプ	イベント	配信するタイミング
IVS Stream Health Change	Starvation Start:	ストリームはストリーマーからデータを受信していません。このストリームの状態を「スタベーション」と呼びます。
IVS Stream Health Change	Starvation End	スタベーションのストリームがストリーマーからデータの受信を開始し、ストリームは健全な状態に戻りました。
IVS Limit Breach	Ingest Bitrate	受信ストリームのビットレートが Amazon IVS の制限を超えています。
IVS Limit Breach	取り込み解像度	受信ストリームの解像度が Amazon IVS の制限を超えています。
IVS Limit Breach	Concurrent Broadcasts	同時にストリーミングされるチャンネルの合計数が Amazon IVS の制限を超えています。
IVS Limit Breach	Concurrent Viewers	同時にチャンネルを視聴している視聴者の合計が Amazon IVS の制限を超えています。

イベントタイプ	イベント	配信するタイミング
IVS Recording State Change	Recording Start	<p>ストリームの処理が開始され、録画プレフィックスが作成され、検証されます。セグメントは、このチャンネル用に設定されたストレージの場所書き込まれます。</p> <p>ライブストリーミングが開始し、Recording Start イベントが発生してから、マニフェストファイルとビデオセグメントがチャンネル用に設定された S3 バケットに書き込まれるまでに少し時間がかかることに注意してください。Recording End イベントの送信後に、録画したストリームを再生または処理することをお勧めします。</p>
IVS Recording State Change	Recording End	<p>ストリームが終了し、このチャンネルの記録が停止します。</p>
IVS Recording State Change	Recording Start Failure	<p>ストリームは開始されましたが、エラー (S3 バケットが存在しない、または正しいリージョンにないなど) により、録画の開始に失敗しました。このライブストリームは録画されません。</p>
IVS Recording State Change	Recording End Failure	<p>録画中に発生したエラー (マスタープレイリストの書き込みの試行が失敗した場合など) により、録画は失敗して終了します。一部のオブジェクトが、設定された保存場所に引き続き書き込まれることがあります。</p>

ストリーム ID に関する注意点: 多くのイベントにある `stream_id` フィールドは、チャンネルがライブになるたびに割り当てられる一意のストリーム識別子です。特定のチャンネルで、各ライブストリームに新しい `stream_id` が付きます。よって、各チャンネルの ARN は、多数の対応するストリーム ID を持つことができます。ストリーム ID を使用すると、同じチャンネル上の異なるストリームセッションを区別できます。

一部のイベントのレイテンシーに関する注意点: エンコーダー構成の設定、特に IDR/キーフレームの間隔は、ストリームの起動のタイミングおよび関連イベントのレイテンシー (Stream Start と Recording Start) に影響します。キーフレームの間隔を短くすると、このレイテンシーが短くなります。IDR/Keyframe の設定の詳細については、Amazon IVS Streaming Configuration の [レイテンシーの低減](#) を参照してください。

Amazon IVS の Amazon EventBridge ルールの作成

Amazon IVS が発行したイベントをトリガーするルールを作成できます。[「Amazon ユーザーガイド」の「Amazon でルールを作成する EventBridge EventBridge」](#) の手順に従います。サービスを選択する際に、[Interactive Video Service (IVS)] を選択します。

例: ストリーム状態の変化

Stream Start: このイベントは、ストリームが処理中で、視聴者に番組を提供できるようになったときに送信されます。

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "IVS Stream State Change",
  "source": "aws.ivs",
  "account": "aws_account_id",
  "time": "2017-06-12T10:23:43Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ivs:us-east-1:aws_account_id:channel/12345678-1a23-4567-
a1bc-1a2b34567890"
  ],
  "detail": {
    "event_name": "Stream Start",
    "channel_name": "Your Channel",
    "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn"
  }
}
```

```
}
```

Stream End: このイベントは、ストリームが処理を停止し、視聴者に番組が提供されなくなったときに送信されます

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "IVS Stream State Change",
  "source": "aws.ivs",
  "account": "aws_account_id",
  "time": "2017-06-12T10:23:43Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ivs:us-east-1:aws_account_id:channel/12345678-1a23-4567-
a1bc-1a2b34567890"
  ],
  "detail": {
    "event_name": "Stream End",
    "channel_name": "Your Channel",
    "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn"
  }
}
```

Stream Failure: このイベントは、処理能力を超えたためストリームが処理されず利用できなくなったときに送信されます。

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "IVS Stream State Change",
  "source": "aws.ivs",
  "account": "aws_account_id",
  "time": "2017-06-12T10:23:43Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ivs:us-east-1:aws_account_id:channel/12345678-1a23-4567-
a1bc-1a2b34567890"
  ],
  "detail": {
    "event_name": "Stream Failure",
    "channel_name": "Your Channel",

```

```
"stream_id": "st-1A2b3c4D5e6F78ghij9Klmn",
"reason": "Transcode capacity exceeded. Please try again."
}
}
```

例: ストリームの正常性の変化

Starvation Start: このイベントは、ストリームがストリーマーからデータを受信していないときに送信されます。このストリームの状態を「スタベーション」と呼びます。

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "IVS Stream Health Change",
  "source": "aws.ivs",
  "account": "aws_account_id",
  "time": "2017-06-12T10:23:43Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ivs:us-east-1:aws_account_id:channel/12345678-1a23-4567-
a1bc-1a2b34567890"
  ],
  "detail": {
    "event_name": "Starvation Start",
    "channel_name": "Your Channel",
    "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn"
  }
}
```

Starvation End: スタベーション状態のストリームがストリーマーからデータの受信を開始し、ストリームが健全な状態に戻ったときに送信されます。

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "IVS Stream Health Change",
  "source": "aws.ivs",
  "account": "aws_account_id",
  "time": "2017-06-12T10:23:43Z",
  "region": "us-east-1",
  "resources": [
```

```
"arn:aws:ivs:us-east-1:aws_account_id:channel/12345678-1a23-4567-
a1bc-1a2b34567890"
],
"detail": {
  "event_name": "Starvation End",
  "channel_name": "Your Channel",
  "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn"
}
}
```

例: 制限違反

すべての制限違反イベントには、違反された制限の名前、制限の値、制限を超過した数 (違反時の値から制限値を引いた値) が含まれます。

Ingest Bitrate: このイベントは、受信ストリームのビットレートが Amazon IVS の制限を超えたときに送信されます。

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "IVS Limit Breach",
  "source": "aws.ivs",
  "account": "aws_account_id",
  "time": "2017-06-12T10:23:43Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ivs:us-east-1:aws_account_id:channel/12345678-1a23-4567-
a1bc-1a2b34567890"
  ],
  "detail": {
    "limit_name": "Ingest Bitrate",
    "limit_value": 1234,
    "exceeded_by": 3,
    "limit_unit": "bits per second",
    "channel_name": "Your Channel",
    "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn"
  }
}
```

Ingest Resolution: このイベントは、受信ストリームの解像度 (総ピクセル数またはエッジあたりのピクセル数) が Amazon IVS の制限を超えたときに送信されます。

最大合計ピクセル数を超過:

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "IVS Limit Breach",
  "source": "aws.ivs",
  "account": "aws_account_id",
  "time": "2017-06-12T10:23:43Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ivs:us-east-1:aws_account_id:channel/12345678-1a23-4567-
a1bc-1a2b34567890"
  ],
  "detail": {
    "limit_name": "Ingest Resolution",
    "limit_value": 495000,
    "exceeded_by": 426600,
    "limit_unit": "total pixels",
    "channel_name": "Your Channel",
    "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn"
  }
}
```

エッジあたりの最大ピクセル数を超過:

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "IVS Limit Breach",
  "source": "aws.ivs",
  "account": "aws_account_id",
  "time": "2017-06-12T10:23:43Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:ivs:us-east-1:aws_account_id:channel/12345678-1a23-4567-
a1bc-1a2b34567890"TBID
  ],
  "detail": {
    "limit_name": "Ingest Resolution",
    "limit_value": 855,
    "exceeded_by": 45,
    "limit_unit": "pixels per edge",
  }
}
```

```
    "channel_name": "Your Channel",
    "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn"
  }
}
```

Concurrent Broadcasts: このイベントは、同時にストリーミングされるチャンネルの合計数が Amazon IVS の制限を超えた場合に送信されます。

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "IVS Limit Breach",
  "source": "aws.ivs",
  "account": "aws_account_id",
  "time": "2017-06-12T10:23:43Z",
  "region": "us-east-1",
  "resources": [],
  "detail": {
    "limit_name": "Concurrent Broadcasts",
    "limit_value": 2,
    "exceeded_by": 3,
    "limit_unit": "active streams"
  }
}
```

Concurrent Viewers: このイベントは、同時にチャンネルを視聴している視聴者の合計が Amazon IVS の制限を超えた場合に送信されます。

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "IVS Limit Breach",
  "source": "aws.ivs",
  "account": "aws_account_id",
  "time": "2017-06-12T10:23:43Z",
  "region": "us-east-1",
  "resources": [],
  "detail": {
    "limit_name": "Concurrent Viewers",
    "limit_value": 10,
    "exceeded_by": 11,
    "limit_unit": "viewers"
  }
}
```

```
}  
}
```

例: 録画状態の変化

すべての録画状態変更イベントについて、このライブストリームのすべてのオブジェクトが保存される最上位パスは `recording_s3_key_prefix` です。失敗した場合、失敗の理由は `recording_status_reason` に示されています。`recording_duration_ms` フィールドは、録画時間のミリ秒数です。

Recording Start: このイベントは、ストリームの処理が開始され、セグメントがチャンネルに設定された保存場所に書き込まれるときに送信されます。

```
{  
  "version": "0",  
  "id": "12345678-1a23-4567-a1bc-1a2b34567890",  
  "detail-type": "IVS Recording State Change",  
  "source": "aws.ivs",  
  "account": "123456789012",  
  "time": "2020-06-23T20:12:36Z",  
  "region": "us-west-2",  
  "resources": [  
    "arn:aws:ivs:us-west-2:123456789012:channel/AbCdef1G2hij"  
  ],  
  "detail": {  
    "channel_name": "Your Channel",  
    "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn",  
    "recording_status": "Recording Start",  
    "recording_status_reason": "",  
    "recording_s3_bucket_name": "r2s3-dev-channel-1-recordings",  
    "recording_s3_key_prefix": "ivs/v1/123456789012/AbCdef1G2hij/2020/6/23/20/12/j8Z9091ndcVs",  
    "recording_duration_ms": 0,  
    "recording_session_id": "a6RfV23ES97iyfoQ"  
  }  
}
```

Recording End: このイベントは、ストリームが終了し、このチャンネルの録画が停止したときに送信されます。

```
{
```

```
"version": "0",
"id": "12345678-1a23-4567-a1bc-1a2b34567890",
"detail-type": "IVS Recording State Change",
"source": "aws.ivs",
"account": "123456789012",
"time": "2020-06-24T07:51:32Z",
"region": "us-west-2",
"resources": [
  "arn:aws:ivs:us-west-2:123456789012:channel/AbCdef1G2hij"
],
"detail": {
  "channel_name": "Your Channel",
  "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn",
  "recording_status": "Recording End",
  "recording_status_reason": "",
  "recording_s3_bucket_name": "r2s3-dev-channel-1-recordings",
  "recording_s3_key_prefix": "ivs/v1/123456789012/AbCdef1G2hij/2020/6/23/20/12/
j8Z9091ndcVs",
  "recording_duration_ms": 99370264,
  "recording_session_id": "a6RfV23ES97iyfoQ",
  "recording_session_stream_ids": ["st-254sopYUvi6F78ghp09vn0A",
"st-1A2b3c4D5e6F78ghij9Klmn"]
}
}
```

Recording Start Failure: このイベントは、ストリームは開始されましたが、エラー (S3 バケットが存在しない、または正しいリージョンにないなど) により、録画の開始に失敗しました。このライブストリームは録画されません。

```
{
  "version": "0",
  "id": "12345678-1a23-4567-a1bc-1a2b34567890",
  "detail-type": "IVS Recording State Change",
  "source": "aws.ivs",
  "account": "123456789012",
  "time": "2020-06-23T20:12:36Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:ivs:us-west-2:123456789012:channel/AbCdef1G2hij"
  ],
  "detail": {
    "channel_name": "Your Channel",
    "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn",
```

```
    "recording_status": "Recording Start Failure",
    "recording_status_reason": "ValidationException",
    "recording_s3_bucket_name": "r2s3-dev-channel-1-recordings",
    "recording_s3_key_prefix": "",
    "recording_duration_ms": 0,
    "recording_session_id": "a6RfV23ES97iyfoQ"
  }
}
```

Recording End Failure: このイベントは、録画中に発生したエラーにより、録画が失敗して終了したときに送信されます。一部のオブジェクトが、設定された保存場所に引き続き書き込まれることがあります。

```
{
  "version": "0",
  "id": "12345678-1a23-4567-a1bc-1a2b34567890",
  "detail-type": "IVS Recording State Change",
  "source": "aws.ivs",
  "account": "123456789012",
  "time": "2020-06-24T07:51:32Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:ivs:us-west-2:123456a7-ab1c-2d34-e5f6-1a2b3c4d5678"
  ],
  "detail": {
    "channel_name": "Your Channel",
    "stream_id": "st-1A2b3c4D5e6F78ghij9Klmn",
    "recording_status": "Recording End Failure",
    "recording_status_reason": "InternalServerError",
    "recording_s3_bucket_name": "r2s3-dev-channel-1-recordings",
    "recording_s3_key_prefix": "ivs/v1/123456789012/AbCdef1G2hij/2020/6/23/20/12/j8Z9091ndcVs",
    "recording_duration_ms": 0,
    "recording_session_id": "a6RfV23ES97iyfoQ"
  }
}
```

AWS CloudTrail を使用した Amazon IVS API コールのログ記録

Amazon Interactive Video Service (IVS) は、Amazon IVS のユーザー、ロール、または AWS のサービスによって実行されたアクションを記録するサービスである AWS CloudTrail と統合されています。CloudTrail は、Amazon IVS のすべての API コールをイベントとしてキャプチャします。キャプチャされるコールには、Amazon IVS コンソールおよびアプリケーションからの API コールが含まれます。

証跡を作成する場合は、Amazon IVS のイベントなど、Amazon S3 バケットへの CloudTrail イベントの継続的な配信を有効にすることができます。証跡を設定しない場合でも、CloudTrail コンソールの [Event history (イベント履歴)] で最新のイベントを表示できます。CloudTrail で収集された情報を使用して、Amazon IVS に対するリクエストや、リクエスト元の IP アドレス、リクエストしたユーザー、リクエスト日時やその他の詳細を確認できます。

CloudTrail の詳細については、[AWS CloudTrail ユーザーガイド](#)を参照してください。

CloudTrail 内の Amazon IVS に関する情報

CloudTrail は、アカウントを作成すると AWS アカウントで有効になります。Amazon IVS でアクティビティが発生すると、そのアクティビティは [Event history (イベント履歴)] の他の AWS のサービスのイベントとともに CloudTrail イベントに記録されます。最近のイベントは、AWS アカウントで表示、検索、ダウンロードできます。詳細については、「[CloudTrail イベント履歴でのイベントの表示](#)」を参照してください。

Amazon IVS のイベントなど、AWS アカウントのイベントの継続的な記録については、証跡を作成します。証跡により、CloudTrail はログファイルを Amazon S3 バケットに配信できます。デフォルトでは、CloudTrail コンソールで作成した証跡が、すべての AWS リージョンに適用されます。証跡では、AWS パーティションのすべてのリージョンからのイベントがログに記録され、指定した Amazon S3 バケットにログファイルが配信されます。さらに、その他の AWS サービスを設定して、CloudTrail ログで収集されたデータを分析し、それに基づく対応を行うことができます。詳細については、CloudTrail ユーザーガイドでこれらの項目を参照してください。

- [お使いの AWS アカウントにおける証跡の作成 \(概要\)](#)
- [CloudTrail のサポート対象サービスと統合](#)
- [Amazon SNSのCloudTrail通知の設定](#)

- [CloudTrail ログファイルを複数のリージョンから受け取る](#)
- [複数のアカウントから CloudTrail ログファイルを受け取る](#)

Amazon IVS のすべてのアクションは CloudTrail によってログ記録されます。関連情報については、「[IVS 低レイテンシーストリーミング API リファレンス](#)」、「[IVS リアルタイムストリーミング API リファレンス](#)」、「[IVS チャット API リファレンス](#)」をご覧ください。例えば、CreateChannel、ListChannels、DeleteChannel の各エンドポイントを呼び出すと、CloudTrail ログファイルにエントリが生成されます。

各イベントまたはログエントリには、リクエストの生成者に関する情報が含まれます。この ID 情報は、リクエストがどのようにして送信されたかを確認するのに役立ちます:

- ルートが使用されたか、AWS Identity and Access Management (IAM) ユーザーの認証情報が使用されたか
- ロールまたはフェデレーテッドユーザーの一時的なセキュリティ認証情報を使用して行われたか。
- 別の AWS のサービスによって行われたか。

詳細については、「[CloudTrail userIdentity エlement](#)」を参照してください。

Amazon IVS ログファイルエントリの概要

証跡は、指定した Amazon S3 バケットにイベントをログファイルとして配信するように設定できます。イベントは任意の発生元からの 1 つのリクエストを表し、リクエストされたアクション、アクションの日時、リクエストのパラメータなどに関する情報が含まれます。

CloudTrail ログファイルには、1 つ以上のログエントリがあります。CloudTrail ログファイルは、公開 API コールの順序付けられたスタックトレースではないため、特定の順序では表示されません。

以下の例は、CreateChannel エンドポイントの CloudTrail ログエントリを示しています。

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "ABCDEFGHIJK1L2EXAMPLE:account_name",
    "arn": "arn:aws:sts::123456789012:assumed-role/First_Streamer/1234567890123456789",
    "accountId": "123456789012",
```

```
"accessKeyId": "ABCDEFGHijkl1EXAMPLE",
"sessionContext": {
  "sessionIssuer": {
    "type": "Role",
    "principalId": "ABCDEFGHijkl1L2EXAMPLE",
    "arn": "arn:aws:iam::123456789012:role/Admin",
    "accountId": "123456789012",
    "userName": "First_Streamer"
  },
  "webIdFederationData": {},
  "attributes": {
    "mfaAuthenticated": "false",
    "creationDate": "2020-04-02T20:57:43Z"
  }
},
"eventTime": "2020-04-02T20:57:46Z",
"eventSource": "ivs.amazonaws.com",
"eventName": "CreateChannel",
"awsRegion": "us-west-2",
"sourceIPAddress": "10.10.10.10",
"userAgent": "console.amazonaws.com",
"requestParameters": {
  "name": "default"
},
"responseElements": {
  "channel": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:channel/1EXAMPLE",
    "authorized": false,
    "ingestEndpoint": "EXAMPLE.global-contribute.live-video.net",
    "latencyMode": "LOW",
    "name": "default",
    "playbackUrl": "https://EXAMPLE.m3u8",
    "tags": {}
  },
  "streamKey": {
    "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/2EXAMPLE",
    "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/1EXAMPLE",
    "tags": {}
  }
},
"requestID": "12a34bc5-EXAMPLE",
"eventID": "a1b2c3de-EXAMPLE",
"readOnly": false,
```

```
"eventType": "AwsApiCall",  
"recipientAccountId": "123456789012"  
}
```

Amazon IVS のセキュリティ

AWS では、クラウドセキュリティが最優先事項です。AWS のお客様は、セキュリティを最も重視する組織の要件を満たすように構築されたデータセンターとネットワークアーキテクチャから利点を得られます。

セキュリティは、AWS とお客様の間の共有責任です。[責任共有モデル](#)では、この責任がクラウドのセキュリティおよびクラウド内のセキュリティとして説明されています。

- クラウドのセキュリティ - AWS は、AWS クラウドで AWS サービスを実行するインフラストラクチャを保護する責任を負います。また、AWS は、使用するサービスを安全に提供します。[AWS コンプライアンスプログラム](#)の一環として、サードパーティーの監査が定期的にセキュリティの有効性をテストおよび検証しています。
- クラウドにおけるセキュリティ - お客様の責任は、使用する AWS のサービスに応じて判断されます。お客様は、データの機密性、組織の要件、および適用法令と規制などのその他要因に対する責任も担います。

このドキュメントは、Amazon IVS を使用する際の責任共有モデルの適用方法を理解するのに役立ちます。以下のトピックでは、セキュリティおよびコンプライアンスの目的を達成するように Amazon IVS を設定する方法について説明します。

トピック

- [データ保護](#)
- [ID とアクセス管理](#)
- [Amazon IVS 用マネージドポリシー](#)
- [Amazon IVS のサービスにリンクされたロールの使用](#)
- [ログ記録とモニタリング](#)
- [インシデントへの対応](#)
- [レジリエンス](#)
- [インフラストラクチャセキュリティ](#)

データ保護

Amazon Interactive Video Service (IVS) に送信されるデータに対して、以下のデータ保護が施されています。

- Amazon IVS は、HTTPS API エンドポイント、RTMPS 取り込み、HTTPS 再生を介して転送中のデータを暗号化します。API エンドポイントの設定は必要ありません。
- 取り込みの場合、ストリーマーは RTMPS を使用してコンテンツを保護できます。これは、デフォルトで利用できます。「[IVS の開始方法](#)」を参照してください。
- IVS チャンネルは、安全ではない RTMP 取り込みを許可するように設定できますが、RTMP を必須とする特定の検証済みのユースケースがない限り、RTMPS の使用をお勧めします。
- トランスコード/トランスマックスの場合、データは内部の Amazon ネットワークで暗号化されずに転送される可能性があります。
- 再生の場合、データは HTTPS 経由で送信されます。
- ライブ動画コンテンツは保存されない、一時的なものです。コンテンツは単にシステム内を移動し、配信中に (内部システムに) キャッシュされます。
- S3 への自動録画機能では、動画コンテンツは Amazon S3 へ書き込まれます。詳細については、[Amazon S3 におけるデータ保護](#)を参照してください。
- 格納されたユーザー入力のメタデータはすべて、サーバー側の暗号化を使用して AWS マネージドサービス内にあります。
- サービス品質を向上させるために、Amazon IVS はお客様 (エンドユーザー) のメタデータ (特定のリージョンのバッファレートなど) を保存しています。このメタデータを、エンドユーザーを個人的に特定するために使用することはできません。
- (ユーザーが管理する) 公開暗号化キーは、ImportPlaybackKeyPair API エンドポイントとともに使用できます。[IVS 低レイテンシーストリーミング API リファレンス](#)を参照してください。これらの暗号化キーを共有しないでください。

Amazon IVS はお客様 (エンドユーザー) データの提供を必要としません。チャンネル、入力、入力セキュリティグループにお客様 (エンドユーザー) データが求められるフィールドはありません。

[Name (名前)] フィールドのような自由記述のフィールドに、お客様 (エンドユーザー) のアカウント番号などの機密扱いの識別情報を入力しないでください。Amazon IVS コンソール、API、AWS CLI、AWS SDK を操作する場合も同様です。Amazon IVS に入力したデータは、診断ログに含まれる場合があります。

ストリームはエンドツーエンドで暗号化されません。ストリームは、IVS ネットワーク内で暗号化されずに送信され、処理されます。

ID とアクセス管理

AWS Identity and Access Management (IAM) は、AWS リソースへのアクセスをアカウント管理者が安全に制御するために役立つ AWS のサービスです。すべての AWS リソースは AWS アカウントによって所有され、リソースの作成またはアクセスは、許可のポリシーによって管理されます。IAM アカウント管理者は、誰を認証 (サインイン) し、誰に Amazon IVS リソースの使用を承認する (アクセス許可を付与する) かを管理します。IAM は追加料金なしで提供される AWS アカウントの機能です。

重要: 詳細については、[AWS IAM 製品ページ](#)、「[IAMユーザーガイド](#)」、「[AWS API リクエストの署名](#)」を参照してください。このセクション全体を通して、IAM ユーザーガイドの特定のセクションへのリンクも提供されています。先に進む前に、この資料を理解しておいてください。

対象者

IAM の用途は、Amazon IVS で行う作業によって異なります。

- サービスユーザー – Amazon IVS サービスを使用して作業を行う場合、管理者から必要なアクセス許可と認証情報が与えられます。さらに多くの Amazon IVS 機能を使用して作業を行う場合は、追加のアクセス許可が必要になることがあります。アクセスの管理方法を理解すると、管理者から適切なアクセス許可をリクエストするのに役に立ちます。Amazon IVS の機能にアクセスできない場合は、[トラブルシューティング](#) を参照してください。
- サービス管理者 – 社内の Amazon IVS リソースの管理を担当している場合は、通常、Amazon IVS へのフルアクセスがあります。管理者は、従業員にアクセスを許可する Amazon IVS 機能とリソースを決定します。その後、サービスユーザーのアクセス許可を変更するために、IAM 管理者にリクエストを送信する必要があります。IAM の基本概念については、このページの情報を確認してください。会社で Amazon IVS を使用して IAM を利用する方法の詳細については、[Amazon IVS で IAM を使用する方法](#) を参照してください。
- IAM 管理者 – IAM 管理者は、Amazon IVS へのアクセスを管理するポリシーを記述できます。IAM で使用できる Amazon IVS アイデンティティベースのポリシーの例は、[アイデンティティベースのポリシーの例](#)を参照してください。

Amazon IVS で IAM を使用する方法

Amazon IVS API リクエストを作成する前に、IAM ID (ユーザー、グループ、ロール) および IAM ポリシーを 1 つ以上作成する必要があります。次に、ポリシーを ID にアタッチします。アクセス許可が反映されるまでに数分かかります。それまでは、API リクエストは拒否されます。

Amazon IVS を IAM と連携させる方法の概要は、「IAM ユーザーガイド」の「[IAM と連携する AWS のサービス](#)」を参照してください。

ID

IAM アイデンティティを作成して、AWS アカウントのユーザーとプロセスに認証を提供できます。IAM グループは、1 つのユニットとして管理できる IAM ユーザーのコレクションです。IAM ユーザーガイドの [ID \(ユーザー、グループ、ロール\)](#) を参照してください。

ポリシー

IAM ユーザーガイドの以下のセクションを参照してください。

- [AWS リソースのアクセス管理](#) — ポリシーに関するすべて
- [Amazon IVS のアクション、リソース、条件キー](#)
- [AWS グローバル条件コンテキストキー](#)
- [IAM JSON ポリシー要素のリファレンス](#) — JSON ポリシーで使用できるすべての要素

デフォルトでは、IAM ユーザーとロールに Amazon IVS リソースを作成または変更するためのアクセス許可はありません (自分のパスワードを変更する場合も含む)。また、AWS コンソール、AWS CLI、または AWS API を使用してタスクを実行することもできません。IAM 管理者は、指定されたリソースで特定の API オペレーションを実行するために必要なアクセス許可をユーザーとロールに付与する、IAM ポリシーを作成する必要があります。

IAM ポリシーは、オペレーションの実行方法を問わず、アクションのアクセス許可を定義します。例えば、iam:GetRole アクションを許可するポリシーがあるとします。このポリシーがあるユーザーは、AWS マネジメントコンソール、AWS CLI、または AWS API からロールの情報を取得できます。

ポリシーは、要素から成る JSON のアクセス許可ポリシードキュメントです。Amazon IVS は 3 つの要素をサポートしています。

- **アクション** — Amazon IVS のポリシーアクションは、アクションの前に `ivs` プレフィックスを使用します。例えば、Amazon IVS `CreateChannel` API メソッドを使用して Amazon IVS チャンネルを作成するアクセス許可を誰かに付与するには、`ivs:CreateChannel` アクションのポリシーにその誰かを含めます。ポリシーステートメントには、`Action` または `NotAction` 要素を含める必要があります。
- **リソース** — Amazon IVS チャンネルリソースには次のような [ARN](#) 形式があります。

```
arn:aws:ivs:${Region}:${Account}:channel/${channelId}
```

例えば、ステートメントで `VgNkEJg0VX9N` チャンネルを指定するには、この ARN を使用します。

```
"Resource": "arn:aws:ivs:us-west-2:123456789012:channel/VgNkEJg0VX9N"
```

リソースの作成など、一部の Amazon IVS アクションは、特定のリソースで実行できません。このような場合は、ワイルドカード (*) を使用する必要があります。

```
"Resource": "*"
```

- **条件** — Amazon IVS は、`aws:RequestTag`、`aws:TagKeys`、および `aws:ResourceTag` のグローバル条件キーをサポートしています。

変数をポリシーのプレースホルダーとして使用できます。例えば、ユーザーの IAM ユーザー名でタグ付けされている場合のみ、リソースにアクセスする IAM ユーザーアクセス許可を付与できます。IAM ユーザーガイドの [変数とタグ](#) を参照してください。

Amazon IVS には、事前設定された一連のアクセス許可をアイデンティティに付与する (読み取り専用またはフルアクセス) ために使用できる AWS マネージドポリシーが用意されています。以下に示す ID ベースのポリシーの代わりにマネージドポリシーを使用することもできます。詳細については、「[Amazon IVS 用マネージドポリシー](#)」を参照してください。

Amazon IVS タグに基づく承認

タグは、Amazon IVS リソースにアタッチすることも、Amazon IVS へのリクエストで渡すこともできます。タグに基づいてアクセスを制御するには、`aws:ResourceTag/key-name`、`aws:RequestTag/key-name`、または `aws:TagKeys` の条件キーを使用して、ポリシーの条件要素でタグ情報を提供します。Amazon IVS リソースのタグ付けの詳細については、「[IVS 低レ](#)

[イテンシーストリーミング API リファレンス](#)」、「[IVS リアルタイムストリーミング API リファレンス](#)」、「[IVS チャット API リファレンス](#)」の「タグ付け」を参照してください。

例については、[タグに基づいて Amazon IVS チャンネルを表示する](#)を参照してください。

ロール

IAM ユーザーガイドの [IAM ロール](#) および [一時的セキュリティ認証情報](#) を参照してください。

IAM ロールは AWS アカウント内のエンティティで、特定の許可を持っています。

Amazon IVS では、一時認証情報の使用をサポートしています。一時的な認証情報を使用して、フェデレーションでサインインする、IAM ロールを引き受ける、またはクロスアカウントロールを引き受けることができます。一時的なセキュリティ認証情報を取得するには、AssumeRole や GetFederationToken などの [AWS Security Token Service](#) API オペレーションを呼び出します。

特権アクセスと非特権アクセス

API リソースには特権アクセスがあります。非特権再生アクセスは、プライベートチャンネルを介してセットアップできます。[プライベートチャンネルの設定](#) を参照してください。

ポリシーのベストプラクティス

IAM ユーザーガイドの [IAM ベストプラクティス](#) を参照してください。

アイデンティティベースのポリシーは非常に強力です。アカウント内で、Amazon IVS リソースを作成、アクセス、削除できるかどうかを決定します。これらのアクションを実行すると、AWS アカウントに追加料金が発生する可能性があります。次の推奨事項に従ってください。

- 最小権限を付与する – カスタムポリシーを作成するときは、タスクの実行に必要なアクセス許可のみを付与します。最小限のアクセス許可から開始し、必要に応じて追加のアクセス許可を付与します。この方法は、寛容なアクセス許可で開始し、後でそれらを強化しようとするよりも安全です。具体的には、ivs:* を管理アクセスのために使用し、アプリケーションでは使用しないでください。
- 機密性の高いオペレーションに Multi-Factor Authentication (MFA) を有効にする – 追加セキュリティとして、機密性の高いリソースまたは API オペレーションにアクセスする IAM ユーザーに対して、MFA を使用するよう求めます。
- 追加のセキュリティとしてポリシー条件を使用する – 実行可能な範囲内で、ID ベースのポリシーでリソースへのアクセスを許可する条件を定義します。例えば、条件を記述して、リクエストが発

生ずる IP アドレスの範囲を指定することができます。指定された日付や時間範囲内でのみリクエストを許可したり、SSL や MFA の使用を要求したりする条件を記述できます。

アイデンティティベースのポリシーの例

Amazon IVS コンソールを使用する

Amazon IVS コンソールにアクセスするには、AWS アカウントの Amazon IVS リソースに関する詳細をリスト化および表示することを許可する最小限の許可セットが必要です。最小限必要なアクセス許可よりも制限された ID ベースのポリシーを作成すると、そのポリシーをアタッチした ID に対してコンソールが意図したとおりに機能しません。Amazon IVS コンソールへのアクセスを確実にするには、以下のポリシーを ID にアタッチします (IAM ユーザーガイドの[IAM アクセス許可の追加および削除](#)を参照してください)。

次のポリシーの部分は、以下へのアクセスを提供します。

- すべての Amazon IVS API エンドポイント
- ユーザーの Amazon IVS [サービスクォータ](#)
- IVS の S3 への自動録画機能(Low-Latency Streaming) と IVS の合成録画機能 (Real-Time Streaming) に必要な Amazon S3 エンドポイント。
- S3 への自動録画サービスリンクロールの作成
- ライブストリームセッションのメトリクスを取得する Amazon CloudWatch

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "ivs:*",
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Action": [
        "servicequotas:ListServiceQuotas"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ],
}
```

```
{
  "Action": [
    "s3:CreateBucket",
    "s3:DeleteBucketPolicy",
    "s3:GetBucketLocation",
    "s3:GetBucketPolicy",
    "s3:ListAllMyBuckets",
    "s3:PutBucketPolicy"
  ],
  "Effect": "Allow",
  "Resource": "*"
},
{
  "Action": [
    "iam:AttachRolePolicy",
    "iam:CreateServiceLinkedRole",
    "iam:PutRolePolicy"
  ],
  "Effect": "Allow",
  "Resource": "arn:aws:iam::*:role/aws-service-role/ivs.amazonaws.com/
AWSServiceRoleForIVSRecordToS3*"
},
{
  "Action": [
    "cloudwatch:GetMetricData"
  ],
  "Effect": "Allow",
  "Resource": "*"
},
{
  "Action": [
    "lambda:AddPermission",
    "lambda:ListFunctions"
  ],
  "Effect": "Allow",
  "Resource": "*"
}
]
```

ユーザーに自分のアクセス許可の表示を許可

この例では、ユーザー ID にアタッチされたインラインおよび管理ポリシーの表示を IAM ユーザーに許可するポリシーを示します。このポリシーには、AWS コンソールで、または AWS CLI もしくは AWS API を使用してプログラマ的に、このアクションを完了するための許可が含まれています。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsWithUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": [
        "arn:aws:iam:*:*:user/${aws:username}"
      ]
    },
    {
      "Sid": "NavigateInConsole",
      "Effect": "Allow",
      "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
      ],
      "Resource": "*"
    }
  ]
}
```

Amazon IVS チャンネルへのアクセス

ここでは、AWS アカウントの IAM ユーザーに、Amazon IVS チャンネルのうちの 1 つである VgNkEJg0VX9N へのアクセス権を付与するとします。また、ストリームの停止 (ivs:StopStream)、メタデータの追加 (ivs:PutMetadata)、チャンネルの更新 (ivs:UpdateChannel) もユーザーに許可します。このポリシーは、Amazon IVS コンソールに必要な ivs:ListChannels、ivs:ListStreams、ivs:GetChannel、および ivs:GetStream のアクセス許可も付与します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ListChannelsInConsole",
      "Effect": "Allow",
      "Action": [
        "ivs:ListChannels",
        "ivs:ListStreams"
      ],
      "Resource": "arn:aws:ivs:*:*:channel/*"
    },
    {
      "Sid": "ViewSpecificChannelInfo",
      "Effect": "Allow",
      "Action": [
        "ivs:GetChannel",
        "ivs:GetStream"
      ],
      "Resource": "arn:aws:ivs:*:*:channel/VgNkEJg0VX9N"
    },
    {
      "Sid": "ManageChannel",
      "Effect": "Allow",
      "Action": [
        "ivs:StopStream",
        "ivs:PutMetadata",
        "ivs:UpdateChannel"
      ],
      "Resource": "arn:aws:ivs:*:*:channel/VgNkEJg0VX9N"
    }
  ]
}
```

```
}
```

タグに基づいて Amazon IVS チャンネルを表示する

アイデンティティベースのポリシーの条件を使用して、タグに基づいて Amazon IVS リソースへのアクセスをコントロールできます。この例では、チャンネルを表示できるポリシーを示します。このポリシーは、このアクションを Amazon IVS コンソールで完了するために必要なアクセス許可も付与します。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ListWidgetsInConsole",
      "Effect": "Allow",
      "Action": "ivs:ListChannels",
      "Resource": "arn:aws:ivs:*:*:channel/*"
    },
    {
      "Sid": "ViewChannelIfOwner",
      "Effect": "Allow",
      "Action": "ivs:GetChannel",
      "Resource": "arn:aws:ivs:*:*:channel/*",
      "Condition": {
        "StringEquals": {"aws:ResourceTag/Owner": "${aws:username}"}
      }
    }
  ]
}
```

このポリシーをアカウントの IAM ユーザーにアタッチできます。ただしアクセス許可は、ユーザーの所有者としてのユーザー名がチャンネルにタグ付けされている場合にのみ付与されます。richard-roe という名前のユーザーが Amazon IVS チャンネルを表示しようとした場合、チャンネルに Owner=richard-roe または owner=richard-roe がタグ付けされている必要があります。それ以外の場合、アクセスは拒否されます。(条件キー名では大文字と小文字は区別されないため、条件タグキー Owner は Owner と owner 両方に一致します。)

トラブルシューティング

次の情報は、Amazon IVS と IAM の利用に伴って発生する可能性がある一般的な問題の診断や修正に役立ちます。

- Amazon IVS でアクションを実行する権限がない。

次のエラー例は、mateojackson IAM ユーザーが AWS コンソールを使用してチャンネルの詳細を表示する際に、ivs:GetChannel 許可を持っていない場合に発生します。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
ivs:GetChannel on resource: arn:aws:ivs:us-west-2:123456789012:channel/VgNkEJg0VX9N
```

この場合、Mateo は、ivs:GetChannel アクションを使用して arn:aws:ivs:us-west-2:123456789012:channel/VgNkEJg0VX9N リソースにアクセスできるように、ポリシーの更新を管理者に依頼します。

- アクセスキーを表示する場合

IAM ユーザーアクセスキーを作成した後は、いつでもアクセスキー ID を表示できます。ただし、シークレットアクセスキーを再表示することはできません。シークレットアクセスキーを紛失した場合は、新しいキーペアを作成する必要があります。アクセスキーは 2 つの部分で構成されています。

- アクセスキー ID (AKIAIOSFODNN7EXAMPLE など)
- シークレットアクセスキー (wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY など)

ユーザー名とパスワードと同様、リクエストを認証するために、アクセスキー ID とシークレットアクセスキーの両方を使用する必要があります。ユーザー名とパスワードと同様に、アクセスキーは安全に管理してください。

重要: [正規ユーザー ID を見つける](#) ためであっても、アクセスキーを第三者に提供しないでください。提供すると、第三者にアカウントへの永続的アクセスを取得される可能性があります。

アクセスキーペアを作成する場合、アクセスキー ID とシークレットアクセスキーを安全な場所に保存するように求めるプロンプトが表示されます。シークレットアクセスキーは、作成時にのみ使用できます。シークレットアクセスキーを紛失した場合、新しいアクセスキーを IAM ユーザーに追加する必要があります。

最大 2 つのアクセスキーを持つことができます。すでに 2 つある場合は、新しいキーペアを作成する前に、いずれかを削除する必要があります。IAM ユーザーガイドの [IAM ユーザーのアクセスキーの管理](#) を参照してください。

- 管理者として Amazon IVS へのアクセスを他のユーザーに許可したい。

Amazon IVS へのアクセスを他のユーザーに許可するには、アクセスを必要とする人またはアプリケーションの IAM エンティティ (ユーザーまたはロール) を作成する必要があります。他のユーザーまたはアプリケーションは、このエンティティの認証情報を使用して AWS にアクセスします。次に、Amazon IVS の適切なアクセス許可を付与するポリシーを、そのエンティティにアタッチする必要があります。

開始するには、IAM ユーザーガイドの [IAM が委任した最初のユーザーおよびグループの作成](#) を参照してください。

- AWS アカウント以外のユーザーに Amazon IVS リソースへのアクセスを許可したい。

他のアカウントのユーザーや組織外のユーザーが、リソースへのアクセスに使用できるロールを作成できます。ロールを引き受けるように信頼されたユーザーを指定することができます。リソーススペースのポリシーまたはアクセスコントロールリスト (ACL) をサポートするサービスの場合、それらのポリシーを使用して、リソースへのアクセスを付与できます。関連情報については、IAM ユーザーガイドの次のセクションを参照してください。

詳細	参照
所有している AWS アカウント全体でリソースへのアクセスを許可する方法	所有している別の AWS アカウントへのアクセス権を IAM ユーザーに提供
第三者の AWS アカウントにリソースへのアクセスを許可する方法	第三者が所有する AWS アカウントへのアクセス権を付与する
ID フェデレーションを介してアクセスを許可する方法	外部で認証されたユーザー (ID フェデレーション) へのアクセスの許可
クロスアカウントアクセスで使用しているロールとリソースベースポリシーの相違点	IAM でのクロスアカウントのリソースへのアクセス

Amazon IVS 用マネージドポリシー

AWS マネージドポリシーは、AWS が作成および管理するスタンドアロンポリシーです。AWS マネージドポリシーは、多くの一般的なユースケースで権限を提供できるように設計されているため、ユーザー、グループ、ロールへの権限の割り当てを開始できます。

AWS マネージドポリシーは、ご利用の特定のユースケースに対して最小特権の権限を付与しない場合があることにご注意ください。AWS のすべてのお客様が使用できるようになるのを避けるためです。ユースケース別に[カスタマーマネージドポリシー](#)を定義することで、アクセス許可を絞り込むことをお勧めします。

AWS マネージドポリシーで定義したアクセス権限は変更できません。AWS が AWS マネージドポリシーに定義されている権限を更新すると、更新はポリシーがアタッチされているすべてのプリンシパルアイデンティティ (ユーザー、グループ、ロール) に影響します。新しい AWS のサービスを起動するか、既存のサービスで新しい API オペレーションが使用可能になると、AWS が AWS マネージドポリシーを更新する可能性が最も高くなります。

詳細については、「IAM ユーザーガイド」の「[AWS マネージドポリシー](#)」を参照してください。

IVSReadOnlyAccess

[IVSReadOnlyAccess](#) AWS マネージドポリシーを使用して、アプリケーション開発者が変更されないすべての IVS API エンドポイント (低レイテンシーとリアルタイムストリーミングの両方) にアクセスできるようにします。

IVSFullAccess

[IVSFullAccess](#) AWS マネージドポリシーを使用して、ユーザーがすべての IVS Chat API エンドポイント (低レイテンシーとリアルタイムストリーミングの両方) にアクセスできるようにします。このポリシーには、IVS コンソールへのフルアクセスを許可するための依存サービスの追加権限が含まれています。

ポリシー更新

Amazon IVS がこれらの変更の追跡を開始したため、このサービスの AWS マネージドポリシーにおける更新情報の詳細を確認します。このページへの変更に関する自動アラートを受け取るには、Amazon IVS Low-Latency Streaming の[ドキュメント履歴](#)ページで RSS フィードにサブスクライブしてください。

変更	説明	日付
IVSFullAccess – 新しいポリシー	IVS は、IVS (低レイテンシーストリーミングとリアルタイムストリーミングの両方) と IVS Chat へのフルアクセスを許可する新しいポリシーを追加しました。	2023 年 12 月 5 日
IVSReadOnlyAccess – 新しいポリシー	IVS は、IVS (低レイテンシーとリアルタイムストリーミングの両方) への読み取り専用アクセスを許可する新しいポリシーを追加しました。	2023 年 12 月 5 日
Amazon IVS が変更の追跡を開始しました。	Amazon IVS が AWS マネージドポリシーの変更の追跡を開始しました。	2023 年 12 月 5 日

Amazon IVS のサービスにリンクされたロールの使用

Amazon IVS は、IAM [サービスにリンクされたロール](#)を使用しています。サービスにリンクされたロールは、AWS のサービスに直接リンクされた一意のタイプの IAM ロールです。サービスリンクロールは Amazon IVS によって事前に定義されており、サービスがユーザーに代わって AWS のその他サービスを呼び出すために必要なすべての許可が含まれています。

サービスにリンクされたロールを使用することで、必要なアクセス権限を手動で追加する必要がなくなるため、Amazon IVS の設定が簡単になります。Amazon IVS は、サービスにリンクされたロールの許可を定義します。そのロールを引き受けることができるのは Amazon IVS のみです。定義した許可には、信頼ポリシーと許可ポリシーが含まれます。この許可ポリシーを他の IAM エンティティにアタッチすることはできません。

IVS サービスリンクロールは、関連する IVS リソースを削除した後でしか削除できません。これにより、サービスにリンクされたロールに関連付けられた AWS リソースに IVS がアクセスするための許可を誤って削除してしまうことを防ぐことができます。

サービスリンクロールをサポートする他のサービスについては、「[IAM と連携する AWS のサービス](#)」でサービスリンクロール列がはいになっているサービスを検索してください。サービスにリンクされたロールに関するドキュメントをサービスで表示するには、リンクではいを選択します。

Amazon IVS のサービスにリンクされたロールのアクセス権限

Amazon IVS では、AWSserviceOREFORIVSRECORDOS3 と呼ばれるサービスにリンクされたロールを使用して、Amazon IVS チャンネルに代わって Amazon S3 バケットにアクセスします。

サービスにリンクされたロール AWSServiceRoleForAccessAnalyzer は、以下のサービスを信頼してロールを引き受けます。

- `ivs.amazonaws.com`

ロールのアクセス許可ポリシーは、指定したリソースに対して以下のアクションを実行することを Amazon IVS に許可します。

- アクション: `your Amazon S3 buckets` 上で `s3:PutObject`

サービスリンクロールの作成、編集、削除を IAM エンティティ (ユーザー、グループ、ロールなど) に許可するには、アクセス許可を設定する必要があります。詳細については、IAM ユーザーガイドの[サービスにリンクされたロールのアクセス権限](#)を参照してください。

Amazon IVS のサービスにリンクされたロールの作成

IVS 用のサービスにリンクされたロールを手動で作成する必要はありません。Amazon IVS コンソール、AWS CLI、または AWS API でレコーディング設定リソースを作成すると、Amazon IVS がサービスにリンクされたロールを作成します。サービスにリンクされたロールの名前は、AWSServiceRoleForIVSRecordToS3 です。

Important

このサービスリンクロールは、このロールでサポートされている機能を使用する別のサービスでアクションが完了した場合にアカウントに表示されます。詳細については、「[IAM アカウントに表示される新しいロール](#)」を参照してください。

サービスにリンクされたこのロールを削除したが、再作成する必要がある場合は、同じプロセスで、アカウントにロールを再作成することができます。録画設定リソースを作成すると、Amazon IVS によってサービスにリンクされたロールが再作成されます。

Amazon IVS のサービスにリンクされたロールの編集

Amazon IVS では、AWSServiceRoleForCloudHSM サービスにリンクされたロールを編集することはできません。サービスにリンクされたロールを作成すると、多くのエンティティによってロールが参照される可能性があるため、ロール名を変更することはできません。ただし、IAM を使用したロールの説明の編集はできます。詳細については、IAM ユーザーガイドの「[サービスにリンクされたロールの編集](#)」を参照してください。

Amazon IVS のサービスにリンクされたロールの削除

サービスにリンクされたロールが必要な機能またはサービスが不要になった場合には、そのロールを削除することをお勧めします。そうすることで、積極的にモニタリングまたは保守されていない未使用のエンティティを排除できます。ただし、手動で削除する前に、サービスにリンクされたロールのリソースをクリーンアップする必要があります。

Note

リソースを削除する際に、Amazon IVS サービスでロールが使用されている場合、削除は失敗することがあります。失敗した場合は、数分待ってから再度オペレーションを実行してください。

AWSServiceRoleForIVSRecordToS3 サービスリンクロールが使用している Amazon IVS リソースを削除するには:

Amazon IVS コンソール、AWS CLI、または AWS API を使用して、すべてのチャンネルから録画設定関連付けを削除し、リージョン内のすべての録画設定リソースを削除します。

IAM を使用してサービスリンクロールを手動で削除するには:

IAM コンソール、AWS CLI、または AWS API を使用して、AWSServiceRoleForIVSRecordToS3 サービスリンクロールを削除します。詳細については、IAM ユーザーガイドの「[サービスにリンクされたロールの削除](#)」を参照してください。

Amazon IVS サービスにリンクされたロールがサポートされるリージョン

Amazon IVS は、サービスを利用できるすべてのリージョンで、サービスにリンクされたロールの使用をサポートします。詳細については、[Amazon IVS サービスエンドポイント](#)を参照してください。

ログ記録とモニタリング

パフォーマンスおよび/またはオペレーションを記録するには、Amazon CloudTrail を使用します。「[AWS CloudTrail を使用した Amazon IVS API コールのログ記録](#)」を参照してください。

インシデントへの対応

インシデントを検出またはアラートするために、Amazon EventBridge イベントを介してストリームの状態をモニタリングできます。「[Amazon IVS 低レイテンシーストリーミングで Amazon EventBridge を使用する](#)」および「[Amazon IVS リアルタイムストリーミングで Amazon EventBridge を使用する](#)」を参照してください。

Amazon IVS の全体的な状態 (リージョン別) に関する情報を得るには、[AWS Health Dashboard](#) を使用します。

レジリエンス

IVS API は、AWS グローバルインフラストラクチャを使用し、AWS リージョンとアベイラビリティゾーンを中心として構築されます。AWSリージョンには、次のような複数のアベイラビリティゾーンがあります。

- 物理的に独立して、隔離されている。
- 低レイテンシー、高スループット、高度の冗長ネットワークで接続されている。
- 従来の単一または複数のデータセンターインフラストラクチャよりも可用性、耐障害性、拡張性に優れている。

API の詳細については、「[IVS 低レイテンシーストリーミング API リファレンス](#)」、「[IVS リアルタイムストリーミング API リファレンス](#)」、「[IVS チャット API リファレンス](#)」を参照してください。AWS リージョンとアベイラビリティゾーンの詳細については、「[AWS グローバルインフラストラクチャ](#)」を参照してください。

Amazon IVS 動画データプレーン

動画の取り込みと配信は、Amazon IVS のコンテンツ配信ネットワーク (CDN) 上で実行されます。CDN は、低レイテンシーの動画専用的高度に調整されています。これにより、Amazon IVS は、エンドツーエンドの高品質の動画を世界中の視聴者に最小限の遅延で配信できます。動画 CDN にはグローバルな POP (Point Of Presence) があり、配信者や視聴者を地理的に分散させることができます。

Amazon IVS リソースを設定するために選択した AWS リージョンに関わらず、次のようになります。

- ストリーマーは、地理的に近い場所にある POP に自動的に動画を取り込みます。
- 視聴者は、グローバル動画 CDN を介して動画をストリーミングします。

取り込まれた動画ストリームは、複数の Amazon IVS データセンターのうちの 1 つで処理され、トランスコードされます。Amazon IVS では、取り込みまたはトランスコーディングの失敗に対する自動フェイルオーバーは提供されません。代わりに、ストリーマーはエンコードまたはブロードキャストクライアントを、配信エラー時に自動的に再取り込みするように設定する必要があります。

インフラストラクチャセキュリティ

マネージドサービスである Amazon IVS は AWS グローバルネットワークセキュリティ手順で保護されています。これらは、[Best Practices for Security, Identity, & Compliance](#) に記載されています。

API 呼び出し

ネットワーク経由で Amazon IVS にアクセスするには、AWS が発行した API 呼び出しを使用します。クライアントは、Transport Layer Security (TLS) 1.2 以降をサポートする必要があります。(以前のバージョンの脆弱性により) TLS 1.3 以降を推奨します。また、一時的ディフィー・ヘルマン Ephemeral Diffie-Hellman (DHE) や Elliptic Curve Ephemeral Diffie-Hellman (ECDHE) などの Perfect Forward Secrecy (PFS) を使用した暗号スイートもクライアントでサポートされている必要があります。これらのモードは、Java 7 以降など、最近のほとんどのシステムでサポートされています。

API リクエストは、アクセスキー ID と、IAM プリンシパルに関連付けられているシークレットアクセスキーを使用して署名する必要があります。または、[AWS Security Token Service](#) を使用して一時的なセキュリティ認証情報を生成し、リクエストに署名することもできます。

これらの API オペレーションは任意のネットワークの場所から呼び出すことができますが、Amazon IVS ではリソースベースのアクセスポリシーがサポートされており、これにはソース IP アドレスに

基づく制限が含まれる場合があります。また、Amazon IVS ポリシーを使用して、特定の Amazon Virtual Private Cloud (Amazon VPC) エンドポイントまたは特定の VPC からのアクセスを管理することもできます。これにより、実質的に AWS ネットワーク内の特定の VPC からの Amazon IVS リソースへのネットワークアクセスが分離されます。

また、すべての API リクエストは sigv4 で署名されます。

API については、「[IVS 低レイテンシーストリーミング API リファレンス](#)」、「[IVS リアルタイムストリーミング API リファレンス](#)」、「[IVS チャット API リファレンス](#)」を参照してください。

ストリーミングおよび再生

再生は HTTPS 経由でエッジから視聴者に行われ、「コントリビューションエッジ」(取り込みエンドポイント)は RTMPS (TLS 経由の RTMP)、または安全でない取り込みを許可するようにチャンネルが構成されている場合は RTMP をサポートしています。Amazon IVS ストリーミングでは、TLS バージョン 1.2 以降が必要です。ストリームはエンドツーエンドで暗号化されません。ストリームは、IVS ネットワーク内で暗号化されずに送信され、処理されます。

サービスクォータ (低レイテンシーストリーミング)

以下は、Amazon Interactive Video Service (IVS) のエンドポイント、リソース、およびその他のオペレーションのサービスクォータと制限です。サービスクォータ (制限とも呼ばれます) とは、AWS アカウントのサービスリソースまたはオペレーションの最大数のことです。つまり、これらの制限は、表に明記されていない限り AWS のアカウントごとに適用されます。[AWS サービスクォータ](#) を参照してください。

AWS のサービスにプログラムで接続するには、エンドポイントを使用します。[AWS サービスエンドポイント](#) も参照してください。

すべてのクォータはリージョンごとに適用されます。

重要 すべてのアカウントで、同時視聴と同時ストリームの数に制限があります。(ビューとは、動画をアクティブにダウンロードまたは再生している固有の視聴セッションです。詳細は[用語集](#)を参照してください)。制限が適切であることを確認し、必要に応じて引き上げをリクエストします (特に大規模なストリーミングイベントを計画している場合)。

サービスクォータの引き上げ

引き上げ可能なクォータについては、[AWS コンソール](#) からレートを引き上げをリクエストできます。コンソールでは、サービスクォータに関する情報も閲覧できます。

API コールレートクォータは調整できません。

API コールレートクォータ

エンドポイントタイプ	エンドポイント	デフォルト値
チャンネル	BatchGetChannel	5 TPS
チャンネル	CreateChannel	5 TPS
チャンネル	DeleteChannel	5 TPS
チャンネル	GetChannel	5 TPS
チャンネル	ListChannels	5 TPS

エンドポイントタイプ	エンドポイント	デフォルト値
チャンネル	UpdateChannel	5 TPS
再生制限ポリシー	CreatePlaybackRestrictionPolicy	5 TPS
再生制限ポリシー	DeletePlaybackRestrictionPolicy	5 TPS
再生制限ポリシー	GetPlaybackRestrictionPolicy	5 TPS
再生制限ポリシー	ListPlaybackRestrictionPolicies	5 TPS
再生制限ポリシー	UpdatePlaybackRestrictionPolicy	5 TPS
¥プライベートチャンネル	DeletePlaybackKeyPair	3 TPS
¥プライベートチャンネル	GetPlaybackKeyPair	3 TPS
¥プライベートチャンネル	ImportPlaybackKeyPair	3 TPS
¥プライベートチャンネル	ListPlaybackKeyPairs	3 TPS
¥プライベートチャンネル	BatchStartViewerSessionRevocation	2 TPS
¥プライベートチャンネル	StartViewerSessionRevocation	10 TPS
録画設定	CreateRecordingConfiguration	3 TPS
録画設定	DeleteRecordingConfiguration	3 TPS
録画設定	GetRecordingConfiguration	3 TPS
録画設定	ListRecordingConfigurations	3 TPS
ストリーム	GetStream	5 TPS
ストリーム	GetStreamSession	5 TPS
ストリーム	ListStreams	5 TPS
ストリーム	ListStreamSessions	5 TPS

エンドポイントタイプ	エンドポイント	デフォルト値
ストリーム	PutMetadata	チャンネルあたり 5 TPS アカウントあたり 155 TPS
ストリーム	StopStream	5 TPS
ストリームキー	BatchGetStreamKey	5 TPS
ストリームキー	CreateStreamKey	5 TPS
ストリームキー	DeleteStreamKey	5 TPS
ストリームキー	GetStreamKey	5 TPS
ストリームキー	ListStreamKeys	5 TPS
タグ	ListTagsForResource	10 TPS
タグ	TagResource	10 TPS
タグ	UntagResource	10 TPS

その他のクォータ

リソースまたは機能	デフォルト値	引き上げ可能	説明
チャンネル	5,000	はい	AWS リージョン あたりの最大チャンネル数。
同時実行ストリーム	100	はい	AWS リージョン ごとに同時にストリーミング可能なチャンネルの最大数。このしきい値を超えると、ストリームは拒否されます。

リソースまたは機能	デフォルト値	引き上げ可能	説明
同時実行ビュー	15,000	はい	AWS リージョン のすべてのチャンネルでライブチャンネルの再生が許可される視聴最大数。(ビューとは、動画をアクティブにダウンロードまたは再生している固有の視聴セッションです。このページの冒頭の「重要」の記載を参照してください。
取り込みビットレート (チャンネル type が BASIC) の場合)	1.5 Mbps または 3.5 Mbps	いいえ	<p>type が BASIC であるチャンネルにストリーミングできる、1 秒あたりの最大ビット数です。</p> <ul style="list-style-type: none"> 入力ビデオ品質が 480p 以下の場合、デフォルトのクォータは 1.5 Mbps です。 入力画質が 480p 以上 1080p 以下の場合、デフォルトのクォータは 3.5 Mbps です。 <p>警告: このしきい値を超えると、ストリームはただちに切断される可能性があります。チャンネルの type の詳細については、Amazon IVS 低レイテンシーストリーミング API リファレンスを参照してください。</p>

リソースまたは機能	デフォルト値	引き上げ可能	説明
取り込みビットレート (チャンネル type が STANDARD) の場合)	8.5 Mbps	いいえ	type が STANDARD であるチャンネルにストリーミングできる、1 秒あたりの最大ビット数です (デフォルト)。警告: このしきい値を超えると、ストリームはただちに切断される可能性があります。チャンネルの type の詳細については、 Amazon IVS 低レイテンシーストリーミング API リファレンス を参照してください。
取り込みビットレート (チャンネル type が ADVANCED_HD の場合)	8.5 Mbps	いいえ	タイプが ADVANCED_HD であるチャンネルにストリーミングできる、1 秒あたりの最大ビット数です。警告: このしきい値を超えると、ストリームはただちに切断される可能性があります。チャンネルの type の詳細については、 Amazon IVS 低レイテンシーストリーミング API リファレンス を参照してください。

リソースまたは機能	デフォルト値	引き上げ可能	説明
取り込みビットレート (チャンネル type が ADVANCED_SD の場合)	8.5 Mbps	いいえ	タイプが ADVANCED_SD であるチャンネルにストリーミングできる、1 秒あたりの最大ビット数です。警告: このしきい値を超えると、ストリームはただちに切断される可能性があります。チャンネルの type の詳細については、 Amazon IVS 低レイテンシーストリーミング API リファレンス を参照してください。
取り込み解像度	1080p (合計 210 万ピクセル、1920 ピクセル/エッジ)	いいえ	チャンネルにストリーミングできる最大解像度 (チャンネル type に関係なく)。関連するしきい値には、合計ピクセルとエッジあたりのピクセルの 2 つがあります。警告: いずれかのしきい値を超えると、ストリームはただちに切断される可能性があります。チャンネルの type の詳細については、 Amazon IVS 低レイテンシーストリーミング API リファレンス を参照してください。
メタデータのペイロード	1 KB	いいえ	PutMetadata リクエストペイロードの最大サイズ (Amazon IVS API)。
再生認証キーペア	3	いいえ	AWS リージョンごとの再生認証キーペアの最大数。

リソースまたは機能	デフォルト値	引き上げ可能	説明
再生制限ポリシー	3	いいえ	AWS リージョンあたりの再生制限ポリシーの最大数。
再生制限ポリシーの国	200	いいえ	再生制限ポリシー-allowedCountries のリストの最大サイズ、つまりポリシーあたりの最大国数。
再生制限ポリシーのオリジン	5	いいえ	再生制限ポリシー-allowedOrigins のリストの最大サイズ。つまり、ポリシーあたりのオリジンの最大数。
再生制限ポリシーのオリジン長	256	いいえ	allowedOrigins 再生制限ポリシーのリスト内のエントリの最大サイズ (文字単位)。
再生トークンのサイズ	2 KB	いいえ	再生の開始に使用される JSON Web Token (JWT) の最大サイズ。
記録設定	20	はい	AWS リージョンあたりの録画設定の最大数。
ストリームキー	1	いいえ	チャンネルごとのストリームキーの最大数。

Service Quotas と CloudWatch 使用状況メトリクスの統合

CloudWatch を使用して、使用状況メトリクスを介して CloudWatch サービスクォータをプロアクティブに管理できます。これらのメトリクスを使用して、現在のサービスの使用状況を CloudWatch グラフやダッシュボードで視覚化できます。Amazon IVS の使用量メトリクスは、Amazon IVS のサービスクォータに対応しています。

CloudWatch Metric Math 関数を使用して、これらのリソースのサービスクォータをグラフに表示できます。また、使用量がサービスクォータに近づいたときに警告するアラームも設定できます。

使用量メトリクスにアクセスするには

1. Service Quotas のコンソールを開きます。 <https://console.aws.amazon.com/servicequotas/>
2. ナビゲーションペインで、[AWS services (AWS のサービス)] を選択します。
3. AWS のサービス一覧から、Amazon Interactive Video Service を探し、選択します。
4. サービスクォータの一覧から、目的のサービスクォータを選択します。新しいページが開き、サービスクォータ/メトリクスに関する情報が表示されます。

または、CloudWatch コンソールからこれらのメトリクスを取得することもできます。[AWS Namespaces (AWS の名前空間)] で、[Usage (使用量)] を選択します。次に、サービスの一覧から [IVS] を選択します (「[Amazon IVS Low-Latency Streaming のモニタリング](#)」を参照してください。)

AWS/使用量 の名前空間の場合、Amazon IVS には次のメトリクスが表示されます。

メトリクス名	説明
ResourceCount	<p>お使いのアカウントで実行されている特定のリソースの数。リソースは、メトリクスに関連付けられたディメンションによって定義されます。</p> <p>有効な統計: 最大 (1 分間に使用されるリソースの最大数)。</p>

次のディメンションは、使用量メトリクスを絞り込むために使用されます。

ディメンション	説明
Service	リソースが含まれる AWS のサービスの名前。有効な値: IVS。
Class	追跡されているリソースのクラス。有効な値: None。
Type	追跡されるリソースのタイプ。有効な値: Resource。
Resource	<p>AWS リソースの名前。有効な値: ConcurrentStreams 、 ConcurrentViews 。</p> <p>ConcurrentStreams および ConcurrentViews 使用状況メトリクスは、「Amazon IVS 低レイテンシーストリーミングのモニタリング」で説明</p>

ディメンション	説明
	されているように、AWS/IVS 名前空間 (None ディメンション) 内のもののコピーです。 ???

使用状況メトリクスの CloudWatch アラームの作成

Amazon IVS 使用量メトリクスに基づいて CloudWatch アラームを作成するには：

1. Service Quotas のコンソールで、上記の説明に従って目的のサービスクォータを選択します。現在、アラームは および に対して ConcurrentStreamsのみ作成できます ConcurrentViews。
2. Amazon CloudWatch アラーム セクションで、 の作成 を選択します。
3. [Alarm threshold (アラームのしきい値)] のドロップダウンリストから、アラーム値として設定する、適用されたクォータ値のパーセンテージを選択します。
4. [Alarm name (アラーム名)] に、アラームの名前を入力します。
5. [Create (作成)] を選択します。

Amazon IVS ストリーミング設定

Amazon Interactive Video Service (IVS) を使用すれば、開発者は世界中の視聴者に簡単に低レイテンシーで動画を配信できます。Amazon IVS ではストリーマーはストリーム制作を処理するだけでよく、その後 Amazon IVS にストリームを送信します。Amazon IVS は、Amazon IVS プレイヤーを使用して視聴者向けの動画処理 (取り込みおよびトランスコーディング)、配信、再生を実行します。

ライブストリーミング向けの豊富なソリューションをご用意しています。複数のカメラ、ビジュアルスイッチャー、グラフィックコンポジットや、さまざまなオーディオミキシング機器を備えたスタジオでも、スマートフォンから初めてのストリーミングを開始する予定でも、同様のコンセプトやエンコーディングパラメータに対処する必要があります。

このドキュメントでは、動画エンコーダーを Amazon IVS にストリーミングする設定について説明します。このドキュメントは、ストリーミング機能をアプリケーションに組み込む開発者を対象とします。

IVS 低レイテンシーストリーミングでは、オーディオのみの入力サポートされていないことに注意してください。

前提条件

「[IVS の開始方法](#)」の手順に従ってチャンネルを作成し、ストリーミングを設定します。このプロセスでは、チャンネル ARN (Amazon リソースネーム) とストリームキーが、ストリームの取り込みと再生のための URL とともに割り当てられます。ストリーミングアプリケーションには、取り込み URL を指定する必要があります。

このドキュメントを読む前に、以下のことを理解しておく必要があります。

- Amazon IVS の基本: 「[IVS 低レイテンシーストリーミングとは](#)」および「[IVS の開始方法](#)」をご覧ください
- Amazon IVS API: [IVS 低レイテンシーストリーミング API リファレンス](#)をご覧ください。

レイテンシーの低減

Amazon IVS の低レイテンシーのストリーミングは、ほとんどのストリーミングアプリケーションと互換性があり、ストリーミングアプリケーションの設定にわずかな変更を加えるだけで済みます。レ

イテンシーを最小限に抑えるには、Amazon IVS プレイヤーを使用する必要があります。サードパーティの HLS ビデオプレーヤーはサポートされていません。Amazon IVS Player SDK のドキュメントを参照してください。

低レイテンシーのストリーミング向けのストリーミングアプリケーションを準備するには、以下の手順を実行します。(注：すべてのストリーミングアプリケーションでこれらのオプションが使用できるわけではありません)。

- ビデオエンコーダーで、IDR/Keyframe を 2 秒間隔 (レイテンシ end-to-end をさらに低くするには 1 秒) に設定します。

IDR/Keyframe は、ストリームの起動のタイミングと関連 EventBridge イベントのレイテンシー (ストリームの開始と録画の開始) に直接影響します。IDR/Keyframe が 2 秒の場合、ストリーム開始のレイテンシーは約 6~7 秒になります。IDR/Keyframe が 1 秒の場合、ストリーム開始のレイテンシーは約 3~4 秒になります。視聴者は、最初のストリーム開始のレイテンシーの発生後のみ、動画を利用できます。動画は Amazon S3 に自動録画されます。

より短い、1 秒のキーフレーム間隔では、QoS のトレードオフが生じます。これにより、Amazon IVS プレイヤーのアダプティブビットレートストリーミング (ABR) が解像度を頻繁に切り替える可能性があります。この場合、セグメントサイズが小さくなるため、ABR チェックがより頻繁に行われます。解像度の切り替えが増加したり、視聴者のネットワークがセグメントを十分な速度でダウンロードできない場合、バッファリングが増加する可能性があります。キーフレーム間隔を 1 秒または 2 秒に決定するとき、これらのトレードオフを考慮してください。

IDR/Keyframe は、5 秒を超える値に設定しないでください。1 秒または 2 秒を使用する場合よりもストリーム開始レイテンシーが長くなることに加えて、IVS は再生用に生成されたすべてのセグメントが IDR/キーフレームで開始されることを保証できません。IDR/キーフレームで開始されないセグメントでは、視聴者が再生を開始する、またはレンディションを変更するときに、デコードエラーや視覚的な歪みが発生する可能性があります。

- 可能な場合は、x264 の設定でエンコーダーをゼロレイテンシーチューニングに設定します。
- バッファサイズ (VBV) がストリームの平均ビットレート (kilobits-per-second) を超えていないことを確認します。

サードパーティによるストリーミング/転送サービスの回避

コンテンツを Amazon IVS に再ストリーミングまたは転送する際に、サードパーティのサービスを使用しないことを強くお勧めします。レイテンシーが増加する原因となります。低レイテンシーを確保するには、Amazon IVS に直接ストリーミングします。

エンコーダー設定

ストリームの取り込み: コーデック、RTMPS、および 443 番ポート

コーデック: Amazon IVS は、動画向けの H.264、オーディオ向けの AAC (LC) をサポートしています。

Amazon IVS は、ソフトウェアやハードウェアのストリーミングに使用される最も一般的かつ安全な取り込みプロトコルである RTMPS (TLS/SSL 接続を介した Real-Time Messaging Protocol) をサポートしています。Amazon IVS でのストリーミングと再生には、TLS バージョン 1.2 以降が必要です。

動画エンコーダーは、アウトバウンドポート 443/TCP に関連付けられた RTMPS プロトコルを使用して Amazon IVS の取り込みに接続する必要があります。接続するには、パスにポートを含む IVS 取り込みサーバーを指定します。

```
rtmps://<IVS-ingest-server>/<IVS-stream-key>
```

例:

```
rtmps://a1b2c3d4e5f6.global-contribute.live-video.net:443/app/<IVS-stream-key>
```

IVS チャンネルは、安全でない RTMP 取り込みを許可するように設定することもできますが、RTMP を必須とする特定の検証済みのユースケースがない限り、RTMPS を使用することをおすすめします。RTMP をストリーミングする場合は、プロトコルが `rtmp://` に設定されていることを確認し、`:443` ポートを削除してください。例:

```
rtmp://a1b2c3d4e5f6.global-contribute.live-video.net/app/<IVS-stream-key>
```

解像度/ビットレート/FPS

ストリームの解像度は、ビットレートとフレームレート (frames-per-second、または FPS) を大きく決定します。以下のガイドラインを使用してください。これらは当社の推奨事項です。以下に示す解像度は横向き (水平×垂直) ですので注意してください。縦向きの場合は逆にします。

	許容可能な画質 (SD) 480p (852x480)	良好な画質 (HD) 720p (1280x720)	高画質 (フル HD) 1080p (1920x1080)
ビットレート	最大 1500 Kbps	最大 4500 Kbps	最大 8500 Kbps

	許容可能な画質 (SD) 480p (852x480)	良好な画質 (HD) 720p (1280x720)	高画質 (フル HD) 1080p (1920x1080)
FPS	30	30 または 60	30 または 60
キーフレーム間隔	2 秒	2 秒	2 秒

ビットレート、FPS、解像度は相互に関連しています。最適な値は状況によって異なるため、複雑な判断となる可能性があります。当社では、上記の値から開始し必要に応じて調整することを最善の方法としてご案内しています。目標とするのは、ストリーミング中のビデオコンポーネントの明確かつスムーズなモーションと、利用可能な帯域幅内での良好な解像度です。フレームレートや解像度を上げると、総合的な動画品質は向上しますが、必然的に帯域幅によって制限されます。

Amazon IVS では、最大 60 FPS のフレームレート (European PAL 25 および 50 標準フレームレートを含む) がサポートされています。適切なビットレート帯域幅があれば、フレームレートが高いほど画質は向上します。用途によっては、低フレームレートでも問題ありません。例えば、防犯カメラに使用する場合などです。

チャンネルタイプ

チャンネルタイプによって、許容解像度とビットレートが決まります。許容可能な入力解像度またはビットレートを超過すると、ストリームがすぐさま切断される可能性があります。

チャンネルには STANDARD、ADVANCED_SD、ADVANCED_HD、BASIC の 4 つのタイプがあります。チャンネルを作成した際のデフォルトタイプは STANDARD です。

チャンネルの種類によって、ビデオは次のようにトランスコードまたはトランスマックスされます。

- STANDARD チャンネルと ADVANCED チャンネルでは、動画がトランスコードされます。元の入力から複数の品質が生成され、視聴者のデバイスとネットワーク状況に最適なエクスペリエンスが自動的に提供されます。トランスコーディングにより、さまざまなダウンロード速度でより高い再生品質が可能になります。
- BASIC チャンネルでは、動画がトランスマックスされます。Amazon IVS は元の入力を視聴者に配信します。

トランスコードされたすべてのチャンネルにはトランスコードプリセットがあり、どのレンディションが作成されるかを決定します。これらは ABR ラダーと考えてください。利用可能なダウンロード帯域幅とビデオ品質のトレードオフとして、視聴エクスペリエンスを最適化できるようにします。

- STANDARD チャンネルにはデフォルトのトランスコードプリセットが 1 つあります。
- ADVANCED チャンネルには 2 つの選択可能なトランスコードプリセットがあります。
 - 制約のある帯域幅配信は、各品質レベルについて STANDARD よりも低いビットレートを使用します。ダウンロード帯域幅が狭い、および/またはシンプルな動画コンテンツ (トーキングヘッドスタイルなど) の場合に使用してください。
 - 高帯域幅の配信は、各品質レベルに高いビットレートを使用します。ダウンロード帯域幅が広い、および/または複雑な動画コンテンツ (フラッシュ、すばやいシーンチェンジなど) の場合に使用してください。これがデフォルトです。

STANDARD チャンネル

STANDARD チャンネルはトランスコードされます。生成されるビデオの最高解像度はフル HD である 1080p です。これがデフォルトのチャンネルタイプです。

- トランスコードプリセット: デフォルトのトランスコードプリセットラダーが 1 つあります。
- 音声: 360p 以下のレンディションでは、音声はトランスコードされます。その他のレンディションでは、元の音声はパススルーされます。

入力解像度と最大ビットレート	ラダーの詳細
1080p60 (8.5 Mbps)	<ol style="list-style-type: none"> 1. 1080p60 (ソースビットレート) 2. 720p60 (3.4 Mbps) 3. 480p30 (1.4 Mbps) 4. 360p30 (0.63 Mbps) 5. 160p30 (0.23 Mbps)
1080p30 (8.5 Mbps)	<ol style="list-style-type: none"> 1. 1080p30 (ソースビットレート) 2. 720p30 (2.4 Mbps) 3. 480p30 (1.4 Mbps) 4. 360p30 (0.63 Mbps) 5. 160p30 (0.23 Mbps)
1080p60 未満、720p60 より大きい (8.5 Mbps)	<ol style="list-style-type: none"> 1. ソースパススルー 2. 720p60 (3.4 Mbps) 3. 480p30 (1.4 Mbps) 4. 360p30 (0.63 Mbps)

入力解像度と最大ビットレート	ラダーの詳細
	5. 160p30 (0.23 Mbps)
1080p30 未満、720p30 より大きい (8.5 Mbps)	<ol style="list-style-type: none"> 1. ソースパススルー 2. 720p30 (2.4 Mbps) 3. 480p30 (1.4 Mbps) 4. 360p30 (0.63 Mbps) 5. 160p30 (0.23 Mbps)
720p60 (8.5 Mbps)	<ol style="list-style-type: none"> 1. 720p60 (3.4 Mbps) 2. 480p30 (1.4 Mbps) 3. 360p30 (0.63 Mbps) 4. 160p30 (0.23 Mbps)
720p30 (8.5 Mbps)	<ol style="list-style-type: none"> 1. 720p30 (2.4 Mbps) 2. 480p30 (1.4 Mbps) 3. 360p30 (0.63 Mbps) 4. 160p30 (0.23 Mbps)
720p30/60 未満、480p30/60 以上 (8.5 Mbps)	<ol style="list-style-type: none"> 1. 480p30 (1.4 Mbps) 2. 360p30 (0.63 Mbps) 3. 160p30 (0.23 Mbps)

ADVANCED-HD チャネル

ADVANCED-HD チャネルはトランスコードされます。生成されるビデオの最高解像度は HD、720p です。

- トランスコードプリセット: 選択可能なトランスコードプリセットラダーが 2 つあります。
- 音声: 360p 以下のレンディションでは、音声はトランスコードされます。その他のレンディションでは、元の音声が入力パススルーされます。

入力解像度と最大ビットレート	ラダーの詳細
720p60 から 1080p60 まで (8.5 Mbps)	トランスコードプリセット: 高帯域幅配信 (デフォルト):

入力解像度と最大ビットレート	ラダーの詳細
	<p>1. 720p60 (3 Mbps)</p> <p>2. 480p30 (1.3 Mbps)</p> <p>3. 360p30 (0.7 Mbps)</p> <p>4. 160p30 (0.27 Mbps)</p> <p>5. 音声のみ (0.08 Mbps)</p> <p>トランスコードプリセット: 制限付き帯域幅配信:</p> <p>1. 720p60 (2.2 Mbps)</p> <p>2. 480p30 (0.8 Mbps)</p> <p>3. 360p30 (0.4 Mbps)</p> <p>4. 160p30 (0.22 Mbps)</p> <p>5. 音声のみ (0.08 Mbps)</p>
720p30 から 1080p30 まで (8.5 Mbps)	<p>トランスコードプリセット: 高帯域幅配信 (デフォルト):</p> <p>1. 720p30 (2.3 Mbps)</p> <p>2. 480p30 (1.3 Mbps)</p> <p>3. 360p30 (0.7 Mbps)</p> <p>4. 160p30 (0.27 Mbps)</p> <p>5. 音声のみ (0.08 Mbps)</p> <p>トランスコードプリセット: 制限付き帯域幅配信:</p> <p>1. 720p30 (1.9 Mbps)</p> <p>2. 480p30 (0.8 Mbps)</p> <p>3. 360p30 (0.4 Mbps)</p> <p>4. 160p30 (0.22 Mbps)</p> <p>5. 音声のみ (0.08 Mbps)</p>

入力解像度と最大ビットレート	ラダーの詳細
720p30/60 未満、480p30/60 より大きい (8.5 Mbps)	<p>ラダーの詳細</p> <p>トランスコードプリセット: 高帯域幅配信 (デフォルト):</p> <ol style="list-style-type: none"> 1. 2.3 Mbps でトランスコードされたソース 2. 480p30 (1.3 Mbps) 3. 360p30 (0.7 Mbps) 4. 160p30 (0.27 Mbps) 5. 音声のみ (0.08 Mbps) <p>トランスコードプリセット: 制限付き帯域幅配信:</p> <ol style="list-style-type: none"> 1. 1.9 Mbps でトランスコードされたソース 2. 480p30 (0.8 Mbps) 3. 360p30 (0.4 Mbps) 4. 160p30 (0.22 Mbps) 5. 音声のみ (0.08 Mbps)
480p30/60 (8.5 Mbps)	<p>トランスコードプリセット: 高帯域幅配信 (デフォルト):</p> <ol style="list-style-type: none"> 1. 480p30 (1.3 Mbps) 2. 360p30 (0.7 Mbps) 3. 160p30 (0.27 Mbps) 4. 音声のみ (0.08 Mbps) <p>トランスコードプリセット: 制限付き帯域幅配信:</p> <ol style="list-style-type: none"> 1. 480p30 (0.8 Mbps) 2. 360p30 (0.4 Mbps) 3. 160p30 (0.22 Mbps) 4. 音声のみ (0.08 Mbps)

ADVANCED-SD チャンネル

ADVANCED-SD チャンネルはトランスコードされます。使用可能なレンディションは入力品質に制限され、アップコンバージョンは行われません。

- トランスコードプリセット: 選択可能なトランスコードプリセットラダーが 2 つあります。
- 音声: 音声はトランスコードされます。

入力解像度と最大ビットレート	ラダーの詳細
480p30/60 から 1080p30/60 まで (8.5 Mbps)	<p>トランスコードプリセット: 高帯域幅配信 (デフォルト):</p> <ol style="list-style-type: none"> 1. 480p30 (1.3 Mbps) 2. 360p30 (0.7 Mbps) 3. 160p30 (0.27 Mbps) 4. 音声のみ (0.08 Mbps) <p>トランスコードプリセット: 制限付き帯域幅配信:</p> <ol style="list-style-type: none"> 1. 480p30 (0.8 Mbps) 2. 360p30 (0.4 Mbps) 3. 160p30 (0.22 Mbps) 4. 音声のみ (0.08 Mbps)

BASIC チャンネル

BASIC チャンネルはトランスマックスされます。1 つのレンディションが生成されます。

- トランスコードプリセット: 無し
- 音声: 音声はトランスコードされます。

入力解像度と最大ビットレート	ラダーの詳細
3.5 Mbps で 480p30/60 より大きく 1080p30/60 以下	ソースエンコーディングパラメータ (ラダーなし)
480p30/60 (1.5 Mbps)	ソースエンコーディングパラメータ (ラダーなし)

ビデオ設定

次の設定が推奨されます。これらは、ほとんどの H.264 動画エンコーディングソフトウェアまたはハードウェア API で利用可能です。

- ビデオエンコーダーで、IDR/Keyframe を 2 秒間隔 (レイテンシ end-to-end をさらに低くするには 1 秒) に設定します。
- H.264 レベル: メイン
- シーン変更: オフ (推奨)
- Chroma のサブサンプル: YUV420P
- CABAC: 推奨
- ColorSpace: BT.709 (HDTVs およびコンピュータディスプレイ間の互換性を最大限に高めるために推奨)。Amazon IVS ビデオトランスコードはパススルーをサポートしています ColorSpace。上級ユーザーは、他の ColorSpace ビデオやフルレンジビデオを使用できます。

オーディオ設定

以下の設定がサポートされています。

- コーデック: AAC (LC)
- ビットレート: 96 Kbps ~ 320 Kbps
- サンプルレート: 44.1 KHz または 48 KHz (本稼働のオーディオフローと一致させることをお勧めします)
- チャンネル: 最大 2 チャンネル - ステレオ (1: モノラルまたは 2: ステレオのオーディオチャンネルをサポート)

VBR ではなく CBR を使用する

エンコーダーのレート制御方法として、VBR (可変 BitRate) ではなく、常に CBR (定数 BitRate) を使用してください。CBR は、ネットワークの固定帯域幅の特性に適しており、クライアントデバイスに対して、予測可能で安定した動画再生を実現します。固定ビットレートでは、視聴者は接続時に処理される画質レベルを簡単に選択できます。

VBR では、シーンの複雑さによっては、ビットレートのスパイクが発生し、動画が Amazon IVS に到達する前にフレームドロップが発生したり、クライアントのプレイヤーでバッファリングされたりする可能性があります。

CBR のみを使用することを強くお勧めします。VBR を使用すると、ストリーム中により多くのバッファリングが発生し、スムーズな再生ができなくなります。

プログレッシブシグナルを使用する

プログレッシブシグナルフローを使用します。本稼働のフローやエンコーディング中は、インターレース方式の動画は避けてください。プログレッシブストリーム信号では、フレーム全体を一度に表示できるため、再生画質が大幅に向上し、インターレース信号を表示する際に生成されるモーションアーティファクトが発生しません。

ネットワークの要件

一定のアップロードストリームを適切に維持できる、安定したインターネット接続が必要です。インターネット接続が不安定になると、視聴者側でストリームの途切れや遅延が発生する可能性があります。

有線接続を使用します。WiFi および TAK 接続は、QoS /packet-queue の優先順位付けが不適切であるため、むらやレイテンシーが発生する可能性があります。可能な限り、ストリーム向けのハード接続を利用してください。

最低限必要な帯域幅よりも 50% 多い帯域幅を割り当てるよう計画してください。動画のビットストリームをエンコードする際のビットレートの変動を補うためのオーバーヘッドが追加されます。

専用のインターネット VLAN を使用してマシンをエンコードします。エンコーダーを別のネットワークで維持することで、トラフィックによる汚染、帯域幅のボトルネック、セキュリティ上の悪影響など、破壊的になり得る影響を防ぐことができます。

クローズドキャプション

IVS はクローズドキャプションをサポートしています。ストリーマーとして視聴者にキャプションを提供したい場合は、動画エンコーダーを使用して、キャプションデータを受け入れられる形式でストリームに埋め込むか、ストリームと一緒に送信する必要があります。

Amazon IVS は、line 21 CEA-708/EIA-608 形式 (608 over 708 と呼ばれます) のキャプションを受け入れます。以下のいずれかの方法を使用して、キャプションを送信できます。

- ATSC A/72 (SEI user_data) で説明されている、動画のエレメンタリーストリームに埋め込まれた CEA-708/EIA-608。この形式は、テレビ放送のエンコーダーで一般的に使用されるものです。
- RTMPS onCaptionInfo スクリプト/AMF0 タグを介して送信される CEA-708/EIA-608。この形式は、Elemental Technologies や Wowza のようなインターネットブロードキャストのエンコーダーやメディアサーバで一般的に使用されるものです。Amazon IVS Player SDK では、1 つの言語をサポートしていますが、マルチトラックキャプションの再生はサポートしていません。

注意: Amazon IVS Player SDK は、CC1 NTSC フィールド 1 のキャプションデータのみをサポートします。マルチトラックキャプションの再生はサポートされていません。

RTMPS 経由で送信する場合、ペイロードには、次の 2 つの要素のペアがある ECMA 配列が含まれている必要があります。

- 文字 708 を含む type という文字列。
- base64 でエンコードされた CEA-708/EIA-608 ペイロードが含まれる data という文字列。

例:

```
00000000 12 00 00 69 00 00 00 00 00 00 00 02 00 0d 6f 6e |...i.....on|
00000010 43 61 70 74 69 6f 6e 49 6e 66 6f 08 00 00 00 02 |CaptionInfo....|
00000020 00 04 74 79 70 65 02 00 03 37 30 38 00 04 64 61 |..type...708..da|
00000030 74 61 02 00 3c 74 51 41 78 52 30 45 35 4e 41 4e |ta..<tQAxR0E5NAN|
00000040 4c 41 50 79 55 72 76 79 55 49 50 79 52 51 50 7a |LAPyUrvyUIPyRQPz|
00000050 49 35 66 7a 73 37 50 7a 76 4c 50 77 67 56 50 7a |I5fzs7PzvLPwgVPz|
00000060 33 36 66 7a 30 34 2f 78 6f 67 50 79 55 4c 2f 38 |36fz04/xogPyUL/8|
00000070 3d 00 00 09 00 00 00 74 |=......t|
```

Elemental の動画エンコーダーを使用する場合は、次のように設定します。

- キャプションの埋め込みを「capture 608 Field 1」に設定します。
- 出力グループに字幕を RTMPS タグ onCaptionInfo として埋め込みます。

詳細については、ブログ記事「[Adding Closed Captions to an Amazon IVS Live Stream](#)」を参照してください。

FFmpeg を使用したストリーミング

FFmpeg は、動画、オーディオ、およびその他のマルチメディアファイルやストリームを処理するための一連の広範なソフトウェアライブラリで構成される無料のオープンソースプロジェクトです。多くのオペレーティングシステムとデバイスで使用できます。

FFmpeg のインストールおよびその他の情報については、[FFmpeg のウェブサイト](#)を参照してください。最新のスタティックビルドを使用します (コンパイルしないでください)。

インストール後、FFmpeg の音声/動画入力ソースを選択します。次のように、利用可能なものを検索できます。

```
ffmpeg -list_devices true -f dshow -i dummy.
```

詳細については、[こちら](#)を参照してください。利用可能な入力ソースと対象とするキャプチャ方法に応じて、選択したデバイスから直接、動画/音声 (埋め込み) をキャプチャし、FFmpeg で信号をエンコードできるはずです。次に例を示します。

- ウェブカメラ - ロジクールの C920 ウェブカメラからの出力をキャプチャします。

```
ffmpeg -f dshow -video_size 1920x1080 -framerate 30 -i video="HD Pro Webcam C920":audio="Microphone (HD Pro Webcam C920)" -c:v libx264 -b:v 6000K -maxrate 6000K -pix_fmt yuv420p -r 30 -s 1920x1080 -profile:v main -preset veryfast -g 120 -x264opts "nal-hrd=cbr:no-scenecut" -acodec aac -ab 160k -ar 44100 -f flv rtmps://<IVS-ingest-server>/<IVS-stream-key>
```

- 動画ファイル - FFmpeg は、多くの動画ファイル形式とキャプチャカードで動作します。MP4 入力に基づくストリーミングの例を次に示します。

```
ffmpeg -re -i input.mp4 -c:v libx264 -b:v 6000K -maxrate 6000K -pix_fmt yuv420p -s 1920x1080 -profile:v main -preset veryfast -force_key_frames expr:gte(t,n_forced*2) -x264opts "nal-hrd=cbr:no-scenecut" -acodec aac -ab 160k -ar 44100 -f flv rtmps://<IVS-ingest-server>/app/<IVS-stream-key>
```

<IVS-ingest-server> および <IVS-stream-key> への入力内容の詳細については、「[IVS の開始方法](#)」のライブストリーミングソフトウェアのセットアップに関する情報を参照してください。

例:

- 取り込みサーバー: `rtmps://jds34ksdg3las.global-contribute.live-video.net/app/`
- ストリームキー: `sk_us-west-2_abcd1234efgh5678ijkl`

Amazon IVS Broadcast SDK を使用してストリーミングする

Amazon IVS Broadcast SDK は、Amazon IVS を使用して Android、iOS、またはウェブアプリケーションを構築するデベロッパー向けのものです。[こちら](#)から、Amazon IVS ユーザーガイド内のブロードキャスト SDK のドキュメントを参照してください。Android、iOS、およびウェブストリーミング用のガイドを含むサブページがあります。Broadcast SDK を使用することで、ビットレート、フレームレート、解像度をカスタマイズできます。

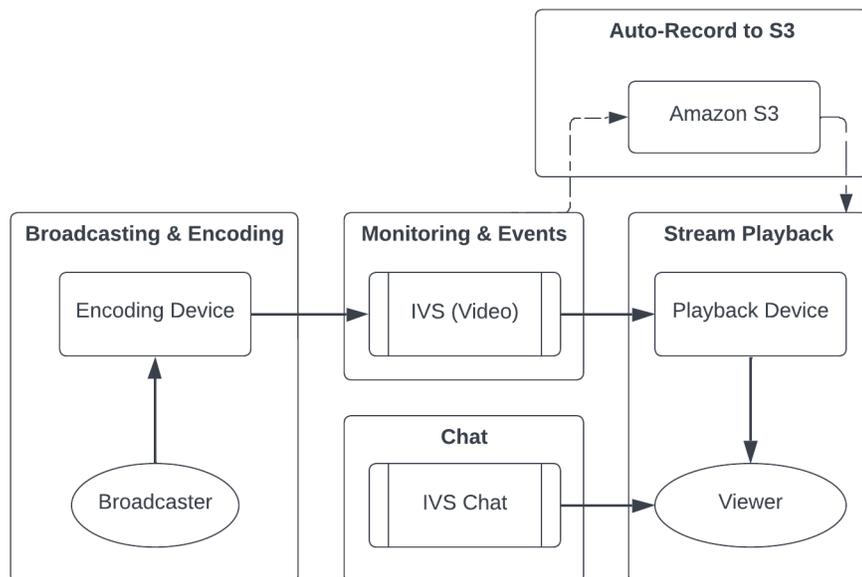
ストリームのテスト

常にストリームの動作を確認します。

[Amazon IVS コンソール](#)で動画ストリームに移動してストリーミングをモニタリングし、ライブストリームを管理します。

トラブルシューティングに関するよくある質問

このドキュメントでは、Amazon Interactive Video Service (IVS) のベストプラクティスおよびトラブルシューティングのヒントを説明します。IVS を使用する際に、予期しない動作や意図しない動作が発生する可能性があります。このような動作は、ブロードキャストからコンテンツの再生まで、ストリーミングプロセスのさまざまな段階で発生する可能性があります。



サポートおよび他の Amazon IVS のリソースについては、「[リソースおよびサポート](#)」を参照してください。

ブロードキャストとエンコーディング

このセクションの質問は、IVS へのストリーミングのブロードキャスト、エンコーディング、およびファーストマイルの条件に関するものです。これらの動作は、コンテンツが IVS サーバーに到達する前に発生します。

トピック:

- [the section called “ストリームスタベーションとは何ですか?”](#)
- [the section called “ストリームが突然停止するのはなぜですか?”](#)
- [the section called “ストリーミング中にネットワークを切り替えるとどうなりますか?”](#)
- [the section called “IVS でマルチリージョンの冗長性を実現するにはどうしたらいいですか?”](#)
- [the section called “IVS Web Broadcast SDK セッションをトラブルシューティングするにはどうすればよいですか?”](#)

- [the section called “Google Chrome の WebRTC-internals メトリクスを使用して IVS Web Broadcast SDK セッションを評価するにはどうすればよいですか?”](#)

ストリームスタベーションとは何ですか？

「ストリームスタベーション」とは、コンテンツを IVS に送信する際、つまり、コンテンツが IVS に取り込まれる際に、コンテンツパケットの配信に生じる遅れや一時停止をいいます。IVS が、エンコーディングデバイスがアドバタイズした想定されたビット数を、一定期間にわたって取り込まなかった場合、これはスタベーション・イベントとみなされます。多くの場合、スタベーションイベントは、ブロードキャスターのエンコーダー、ローカルネットワークの状態、および/またはエンコーディングデバイスと IVS の間のパブリックインターネット経由の転送によって引き起こされます。

視聴者の観点から見ると、スタベーションイベントは、動画における遅延、バッファリング、フリーズとして現れる場合があります。ストリームスタベーションのイベントは、スタベーションのイベントの性質に応じて、短い (5 秒未満) 場合もあれば、長い (数分) 場合もあります。

スタベーションイベントのモニタリングを許可するには、IVS はスタベーションイベントを Amazon EventBridge イベントとして送信 EventBridge します。「[Amazon IVS で Amazon を使用する](#)」の [例: Stream Health Change](#)」を参照してください。これらは、ストリームがスタベーションの状態になったり、スタベーションの状態から脱したりしたときに送信されます。ユースケースによっては、ストリームが断続的な状態であることをブロードキャスターや視聴者に通知するなど、適切な措置を講じることができます。

その他のスタベーションモニタリングツールについては、「[Amazon IVS Low-Latency Streaming のモニタリング](#)」、「IVS [ListStreams](#) API エンドポイント (ヘルスによるフィルタリング)」、「および「IVS [GetStream](#) エンドポイント (個々のストリームの分析)」を参照してください。「[the section called “ストリームスタベーションのイベントをモニタリングするにはどうすればよいですか?”](#)」も参照してください。

ストリームが突然停止するのはなぜですか？

ストリームが突然停止する (ストリームセッションが終了する) 最も一般的な理由は次のとおりです。

- 取り込みデータが欠落している — ストリームセッションの取り込みが 30 秒間完全に停止すると (IVS にデータが取り込まれない)、IVS 取り込みサーバーは IVS ストリームセッションを終了します。この 30 秒間、ブロードキャスターは取り込みサーバーに再接続できます。ただし、場合に

よっては (ネットワークの切り替えなど)、RTMPS の TLS ハンドシェイクが壊れているため、既存のストリームセッションへの再接続ができない場合があります。これに関する一般的な根本原因には、ネットワークの問題 (ブロードキャストデバイスと IVS 間の輻輳など)、ブロードキャストデバイスでのインターネット接続の完全な切断、ブロードキャストデバイスがコンテンツセグメント (FLV タグ) を生成しないなどがあります。

多くの場合、ストリームの切断は、ストリームスタベーションのイベントと連動します。スタベーションのイベントは、データの着信において一時停止が発生した場合にトリガーされます。スタベーション開始イベントが送信されてからストリーム終了イベントが (スタベーション終了イベントなしで) 送信された場合、これは、IVS にデータが送信されなかったためにストリームが終了したことを示すことがよくあります。

- IVS StopStream エンドポイント — IVS ストリームセッション中に [StopStream](#) API コールが行われると、IVS ストリームセッションは終了します。StopStream エンドポイントは、受信した RTMPS ストリームを IVS 取り込みサーバーから切断します。使用しているエンコーディングソフトウェア/ハードウェアによっては、新しいストリームセッションが試行される場合があります。
- エンコーダーエラー — 一部のソフトウェア/ハードウェアエンコーダーは、エンコーディングプロセス中にエラーが発生すると、ストリームセッションを切断します。IVS の観点から見ると、これらの切断は、ブロードキャスターによる意図的な切断として現れます。ただし、エンコーディングログでは、ストリームは意図しないエラーを理由として切断されたと判断される場合があります。

ストリーミング中にネットワークを切り替えるとどうなりますか？

ブロードキャスターがネットワーク (からセルラー WiFi など) を切り替えると、進行中の RTMPS 接続は切断されます。ブロードキャスターのインターネット接続は、おそらく 3~4 秒後に再確立される一方で、ネットワークの切り替えを理由として新しい接続には、新しい IP アドレスが割り当てられ、これにより新しい RTMPS 接続が生成されます。この切り替え中、以前の RTMPS 接続はクリーンに切断されません。エンコーダーは、切断メッセージを IVS に送信しません。その結果、IVS は以前の RTMPS 接続が再接続するまで 30 秒間にわたって待機します。これにより、新しいネットワーク上の新しい RTMPS ストリームによる IVS への接続がブロックされます。

ネットワーク間の切り替えを高速化するには、デバイスがネットワークを切り替えるときに、IVS [StopStream](#) エンドポイントを使用して前のストリームセッションを閉じることをお勧めします。このシナリオでは、ブロードキャストデバイスが新しいネットワークに接続されると、ブロードキャストデバイスは StopStream エンドポイントを呼び出して、現在休止しているストリームを終了できます。StopStream 通話が成功すると、ブロードキャストデバイスは 30 秒間待たずに新しいネットワークで新しいストリームセッションを開始できます。

IVS でマルチリージョンの冗長性を実現するにはどうしたらいいですか？

IVS 内では、いくつかの方法で冗長性を実現できます。IVS のセキュリティの「[耐障害性](#)」を参照してください。

IVS は、コントロールとデータという異なるネットワークプレーンに分かれています。

- コントロールプレーンはリージョンレベルのものであり (AWS リージョンに基づく)、IVS のリソース (チャンネル、ストリームキー、再生キーペア、および録画設定) に関する情報を保存します。
- データプレーンは AWS リージョンに制限されず、取り込みからエグレスまでデータを運ぶネットワークです。例えば、us-west-2 リージョンでチャンネルが作成されたとしても、そのチャンネルにストリーミングされる動画は us-west-2 を経由しない場合があります。

「[グローバルソリューション、リージョナルコントロール](#)」も参照してください。次の 2 つのシナリオを考えてみましょう。

- コントロールプレーンリージョンが 1 つだけ使用されている場合 (例: us-east-1) — 特定の AWS コントロールリージョンでパフォーマンスの低下や停止が発生すると、IVS コントロールプレーンでは、チャンネル、ストリームキー、再生キーペア、録画設定のいずれかを作成、読み取り、更新、削除する際にレイテンシーやエラーが発生する可能性があります。障害発生中に新しいストリームを開始しようとする、ストリームセッションを開始する際にレイテンシーがより長くなったり、エラーが発生したりする可能性があります。パフォーマンス低下の程度によっては、既に進行中のストリームがあるチャンネルへのブロードキャストを続行できる場合があります。

[再生承認](#)が有効になっている場合、現在の視聴者はおそらく進行中のストリームの再生を続行できますが、新しい視聴者は、再生キーペアの承認に問題がある場合には視聴を開始できない可能性があります。再生承認が有効になっていない場合は、現在の視聴者と新しい視聴者の両方が進行中のストリームを視聴できるはずですが、

障害が発生した場合、IVS の S3 への自動録画機能が中断されることもあります。

IVS コントロールプレーンは、リージョンレベルの障害が発生しても、別の AWS リージョンに自動的にフェイルオーバーしません。

- 2 つのコントロールプレーンリージョン (例: us-east-1 と us-west-2) が使用されており、かつ、2 つ目のリージョンがプライマリリージョンが使用できないときのフェイルオーバーである場合 — IVS は、リージョンレベルのコントロールプレーンのフェイルオーバーをネイティブにサポートしていません。したがって、コントロールプレーンのリージョンで問題が発生した場合、新しいスト

リームの開始やコントロールプレーンに対する呼び出しで問題が発生する可能性があります。ただし、データプレーンはおそらく影響を受けないため、コントロールプレーンのリージョンでの進行中のストリームは問題なく続行されます。コントロールプレーンをセカンダリ (フェイルオーバー) リージョンに移行する操作は、アプリケーション側で実行する必要があります。コントロールプレーンのフェイルオーバーを処理するカスタム実装ロジックを記述できます。リージョンレベルのチャンネルのフェイルオーバーの管理方法に関する公式ガイダンスはありません。

動画データプレーンとリージョンレベルのコントロールプレーンを分離することで、IVS アーキテクチャの耐障害性が向上します。リージョンレベルのコントロールプレーンに障害が発生した場合でも、進行中のライブストリームはほとんど、またはまったく中断されません。IVS は 99.9% の連続稼働時間の SLA を維持し、お客様のためにインフラストラクチャの安定性を確保することに取り組んでいます (当社の「[SLA](#)」を参照してください)。

IVS Web Broadcast SDK セッションをトラブルシューティングするにはどうすればよいですか？

[IVS Web Broadcast SDK](#) の動作は、通常の IVS RTMPS 取り込みセッションとは少し異なります。Web Broadcast SDK は、WebRTC プロトコルを利用して IVS エンドポイントにストリーミングします。コンテンツが IVS エンドポイントに入ると、表示用に処理されて、HLS 出力に再多重化/トランスコードされます。

Web Broadcast SDK の性質上、エンコード動作のトラブルシューティングについては次のヒントに注意してください。

- ブロードキャストセッション中に開く必要のないブロードキャストデバイスのタブ/プログラムを閉じてください。余分なタブ/プログラムがコンピューティングリソース (CPU、RAM、ネットワークなど) を使用する可能性があり、ブロードキャストアプリケーションのパフォーマンスが低下する可能性があります。閉じることができないタブ/プログラムについては、コンピューティングリソースを不必要に使用していないことを確認してください。
- デバイスのアップロード速度が 200 Kbps を超えているようにしてください。(これは Web Broadcast SDK の「[既知の問題](#)」の 1 つに記載されています。) アップロード速度を評価するには、ブロードキャストデバイスのタスクマネージャーを開いて、ストリーミング時に利用できるネットワークを分析します。アップロード速度/ビットレートが想定または希望よりも低い場合は、帯域幅を消費している可能性のある他のタブ/プロセスを評価します。また、大量の帯域幅を消費している可能性があるローカルネットワーク上の他のマシンを調べます。
- CPU 使用率にランダムな急上昇がある場合は、マシンのタスクマネージャーを見て、CPU を消費している可能性があるプロセスを確認します。CPU 使用率をランダムに上昇させる一般的なサー

ビスとして、マシン上で定期的なスキャンを実行するウイルス対策ソフトウェアを挙げることでできます。

- <https://stream.ivs.rocks/> 経由でストリーミングして、環境を分離し、アプリケーションロジックが望ましくない動作を引き起こしていないことを確認してみてください。このサイトは IVS によって運営されており、Web Broadcast SDK との統合のいずれかの部分が望ましくない動作の根本原因となっていないかどうかを評価するための堅牢なテスト環境として提供されています。
- Google Chrome の WebRTC-internals を使用してみてください (以下を参照)。

Google Chrome の WebRTC-internals メトリクスを使用して IVS Web Broadcast SDK セッションを評価するにはどうすればよいですか？

IVS Web Broadcast SDK を介してストリーミングする場合、ブロードキャストのエンコードおよび送信中にさまざまな動作が発生する可能性があります。ブロードキャストデバイスのセッションに関する情報をトラブルシューティングまたは収集するには、次の手順に従います。

1. Google Chrome で、ブロードキャストウェブページを開きます。
2. 新しい Chrome タブを開き、`chrome://webrtc-internals/` (これを正確にコピーしてください) に移動します。
3. 元のブロードキャストウェブページのタブで、Web Broadcasting SDK セッションを開始し、動作が観察されるまでセッションを実行します。
4. 動作が確認されたら、`chrome://webrtc-internals/` タブに切り替えて (ブロードキャストセッションは終了しないでください)、正しいウェブページが表示されることを確認します。

▶ Create Dump

Read stats From: **Note:** computed stats are in []. Experimental stats are marked with an * at the end and do not show up in the getStats result.<https://stream.ivs.rocks/> [rid: 3067, lid: 1, pid: 32946]

GetUserMedia Requests

```
https://stream.ivs.rocks/, { iceServers: [], iceTransportPolicy: all, bundlePolicy: max-bundle, rtcpMuxPolicy: require, iceCandidatePoolSize: 0 }
```

```
ICE connection state: new
Connection state: new
Signaling state: new
ICE Candidate pair: (not connected)
▶ ICE candidate grid
```

Stats Tables

Filter statistics by type including

- ▶ certificate (id=CF9C:62:D5:A8:03:45:55:A5:00:F7:0A:59:1D:AA:23:46:DE:31:45:AE:A2:48:6A:03:66:FC:2B:81:2F:2B:32:AD)
- ▶ data-channel (id=D1)
- ▶ track (id=DEPRECATED_TO1)
- ▶ track (id=DEPRECATED_TO2)
- ▶ local-candidate (candidateType=host, tcpType=active, id=l8m+mV7dh)
- ▶ local-candidate (candidateType=host, id=I9P+Kok6N)
- ▶ local-candidate (candidateType=host, id=IUcCiN2O)
- ▶ local-candidate (candidateType=host, tcpType=active, id=ISkSIIGsx)
- ▶ local-candidate (candidateType=host, id=IUUWMOuTJ)
- ▶ local-candidate (candidateType=host, tcpType=active, id=lwKxg6czL)
- ▶ outbound-rtp (kind=audio, mid=1, ssrc=1134012001, id=OT01A1134012001)
- ▶ outbound-rtp (kind=video, mid=0, ssrc=3966401599, id=OT01V3966401599)
- ▶ peer-connection (id=P)
- ▶ media-source (kind=audio, id=SA2)
- ▶ media-source (kind=video, id=SV1)
- ▶ transport (id=T01)
- ▶ Stats graphs for track (id=DEPRECATED_TO1)
- ▶ Stats graphs for outbound-rtp (kind=audio, mid=1, ssrc=1134012001, id=OT01A1134012001)
- ▶ Stats graphs for outbound-rtp (kind=video, mid=0, ssrc=3966401599, id=OT01V3966401599)
- ▶ Stats graphs for peer-connection (id=P)
- ▶ Stats graphs for media-source (kind=audio, id=SA2)
- ▶ Stats graphs for media-source (kind=video, id=SV1)

5. 画面の最上部にある [ダンプを作成] の展開可能なセクションを開きます。
6. 画面の上部 (ダンプの作成の右下) にある PeerConnection 更新と統計データをダウンロードを選択し、関連するセッションから .txt ファイルをダウンロードします。
7. ダウンロードが完了すると、ファイルには WebRTC 接続の履歴ビューが表示されます。これをさまざまなツールで表示したり、さらなる分析のために AWS サポートチームに送信したりできます。

モニタリングとイベント

このセクションの質問は、IVS のモニタリング、メトリクス、およびイベントに関するものです。

トピック:

- [the section called “ストリームスタベーションのイベントをモニタリングするにはどうすればよいですか?”](#)

- [the section called “Amazon を使用して IVS サービスクォータ CloudWatch をモニタリングするにはどうすればよいですか?”](#)
- [the section called “IVS Stream Health を使用してストリームの不安定性を診断するにはどうすればよいですか?”](#)

ストリームスタベーションのイベントをモニタリングするにはどうすればよいですか?

ストリームスタベーションのイベントについては、次のモニタリング方法をお勧めします。

- [Amazon IVS EventBridge を使用した Amazon](#) — ストリームスタベーションイベントが開始または終了すると、IVS は EventBridge ストリームヘルス変更イベントを生成します。Amazon の EventBridge ターゲットとルールを使用すると、これらのストリームスタベーションイベントを使用して、ストリームスタベーションが発生しているときにアラートを受け取ることができます。ターゲットとルールの詳細については、「[Amazon EventBridge ユーザーガイド](#)」を参照してください。
- [Amazon IVS Low-Latency Streaming のモニタリング](#) — ライブストリームセッション中、データが記録され、IVS ストリームのヘルスの分析を介して利用可能になります。これには、エンコーダーの設定、取り込みメトリクス、およびストリームセッションイベントに関する情報が含まれます。これは、進行中のストリームをモニタリングしたり、ストリームを遡及的に評価したりする場合に役立ちます。IVS コンソールまたは API を使用して、スタベーションが発生したストリームを特定できます。ストリームセッションデータは、チャンネルが削除された後も 60 日間利用できるため、スタベーションイベントが発生した過去のストリームを特定するのに役立ちます。
- ヘルスによるストリームのフィルタリング — IVS コンソールまたは IVS [ListStreams](#) API エンドポイントでは、health フィルターを使用して STARVING 状態のストリームセッションを検索できます。また、の IVS CloudWatch メトリクス ConcurrentStreams には、ストリームスタベーション状態にあるストリームの合計数を収集するために使用できる Health デイメンションが含まれています。「[Amazon IVS Low-Latency Streaming のモニタリング](#)」を参照してください。
- IVS [GetStream](#) エンドポイントを使用して、個々のストリームを分析できます。

「[the section called “ストリームスタベーションとは何ですか?”](#)」も参照してください。

Amazon を使用して IVS サービスクォータ CloudWatch をモニタリングするにはどうすればよいですか？

Amazon を使用して、IVS サービスクォータ CloudWatch をプロアクティブにモニタリング/管理できます。「[IVS Service Quotas](#)」を参照してください。このドキュメントには、使用状況メトリクスのアラームの作成 CloudWatchに関する情報が含まれています。

アラームがトリガーされたときに適切な個人/グループに通知する適切な SNS トピックを設定することをお勧めします。アラームがトリガーされ、クォータが引き上げ可能な場合は、新しい値でサービスクォータの引き上げをリクエストする必要があります。引き上げのリクエストについては、「[IVS Service Quotas](#)」を参照してください。

IVS Stream Health を使用してストリームの不安定性を診断するにはどうすればよいですか？

IVS Stream Health ダッシュボードを使用してストリームの不安定性を評価することをお勧めします。手順については、「[Amazon IVS Low-Latency Streaming のモニタリング](#)」を参照してください。

ダッシュボードには、動画のビットレート、フレームレート、音声のビットレートの時系列グラフがあります。以下に例を示します。また、「[で表示 CloudWatch](#)」をクリックして、Amazon でデータを表示することもできます CloudWatch。

以下では、いくつかのシナリオを説明します。

インターネット帯域幅が低いか、インターネットが輻輳している

この場合、ビットレートを低くしても、ストリームは比較的不安定です。ブロードキャスターと ISP の間、もしくは ISP と IVS の間に十分な帯域幅がないか、または IVS へのネットワークパスに問題があります。これを解決するには、他のネットワークプロセスが帯域幅を使用していないことを確認するか、または ISP にネットワーク診断を依頼してください。

IVS Stream Health ダッシュボード:

Video bitrate



Frame rate



Audio bitrate



CloudWatch:

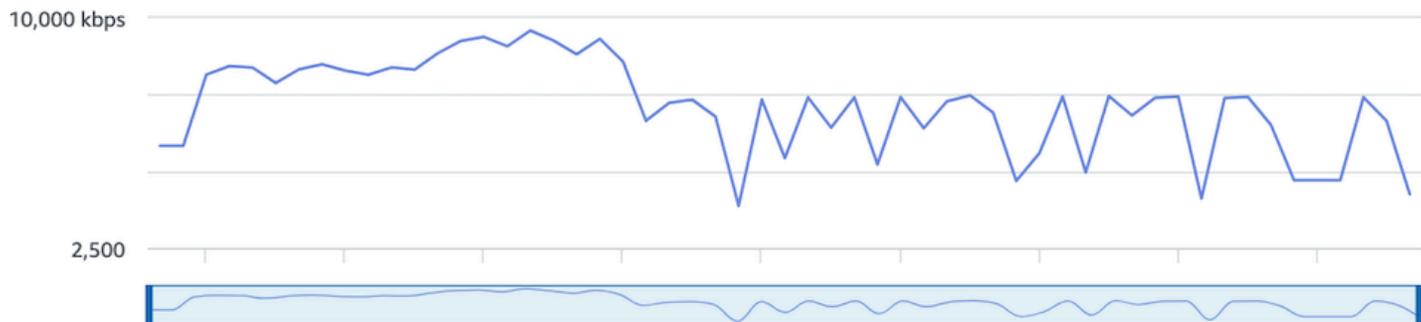


ビットレートが過度に高い

ビットレートが高ければ高いほど品質が良いとは限りません。ここでは、ビットレートが高いことで不安定になっています。ネットワークの輻輳により、高いビットレートでのブロードキャストが、ストリームの不安定さを引き起こすケースが多くあります。「[the section called “解像度/ビットレート/FPS”](#)」に記載されている最大ビットレートを遵守してください。

IVS Stream Health ダッシュボード:

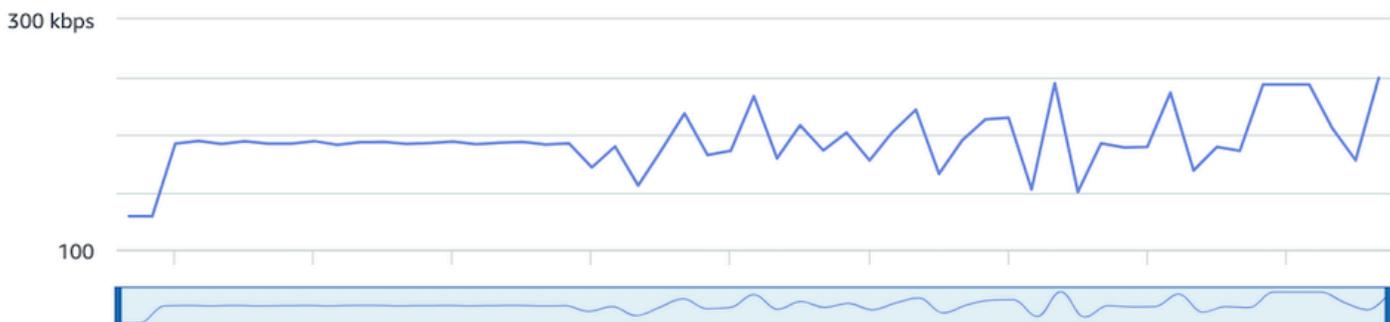
Video bitrate



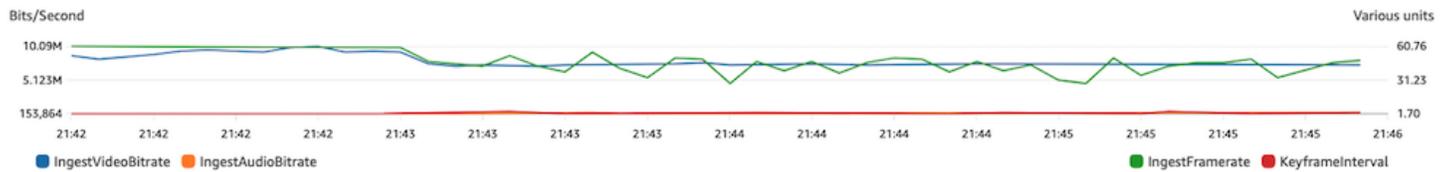
Frame rate



Audio bitrate



CloudWatch:

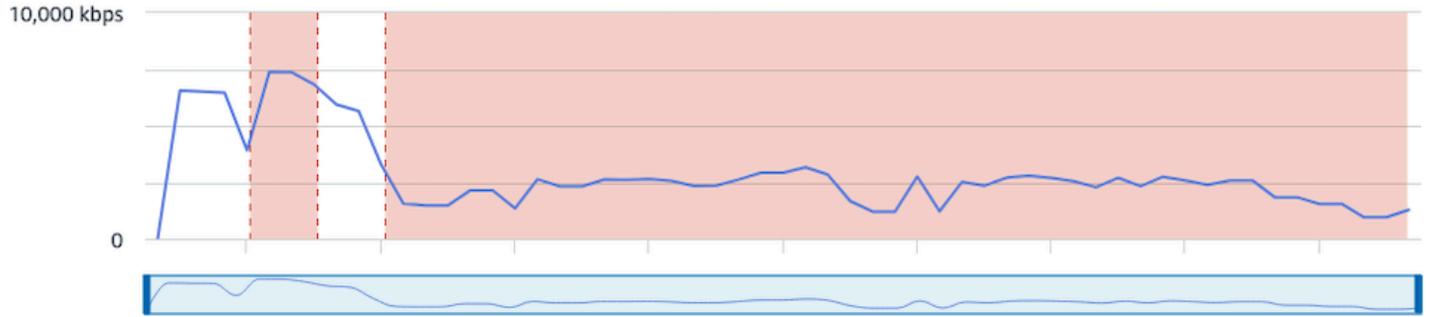


ネットワークまたはハードウェアの問題

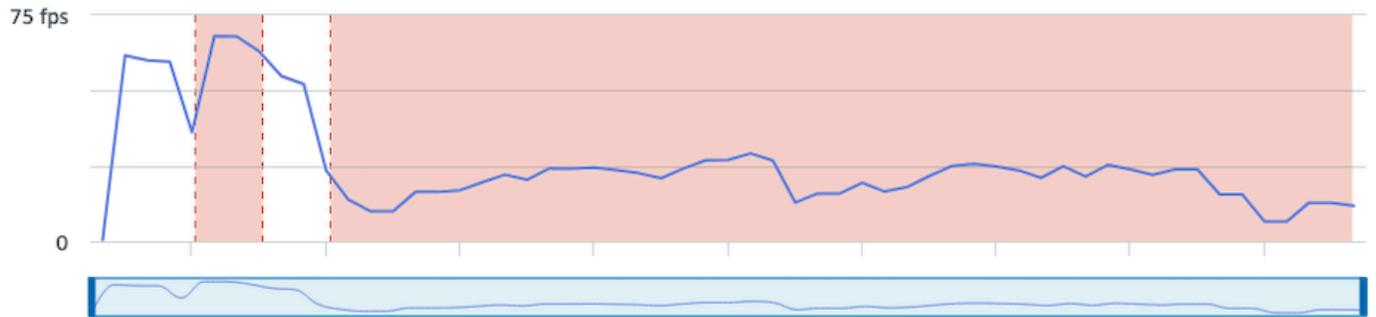
動画エンコーディングには多くのコンピューティングリソースが必要であり、動画エンコーディングを実行しているマシンが負荷に追いつけない場合があります。この場合は、マシンが過負荷 (同時に実行する処理が多すぎる状態) となっていないこと、およびエンコーダーが最新であることを確認してください。CPU 使用量のより少ないエンコーディングプリセットに切り替えることを検討してください。

IVS Stream Health ダッシュボード:

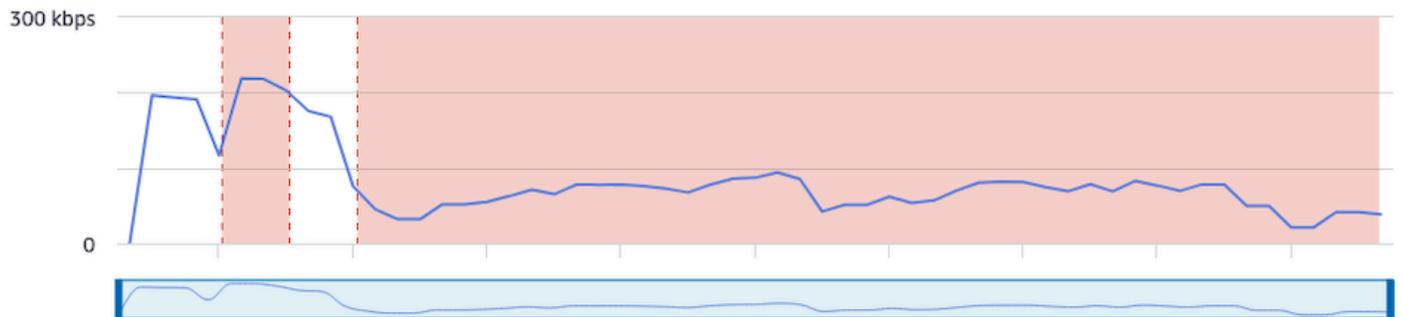
Video bitrate



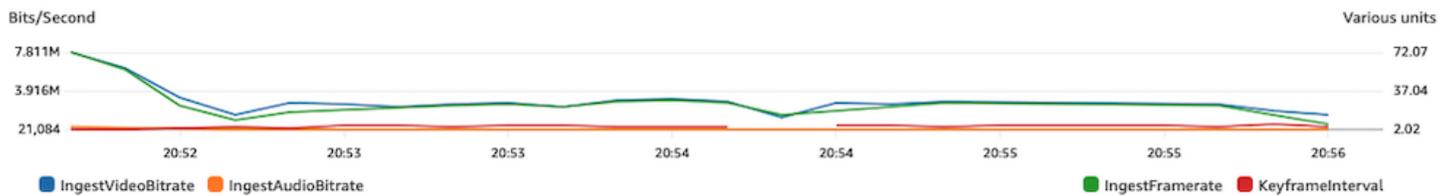
Frame rate



Audio bitrate



CloudWatch:

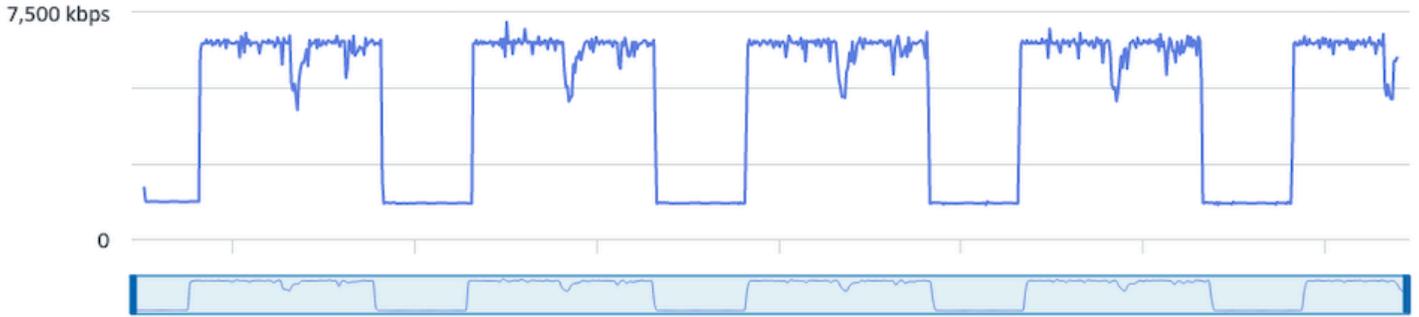


ビットレートの急増と急減

ストリーミングエンコーダーは、圧縮するフレームの複雑さに応じて、過度にビットレートを最適化しようとする場合があります。ビットレートが急激に変動する場合に、視聴者が多すぎるデータをロードしようとする、バッファリングが発生する可能性があります。固定ビットレート (CBR) が有効になっているようにしてください。これにより、フレームの複雑さにかかわらず、ストリーム全体で一貫したビットレートが維持されます。ビットレートの急減が発生することもあるので注意してください。これは、エンコーダーによる動画の圧縮のために十分な CPU 性能をマシンが備えていないことを示唆している可能性があります。

IVS Stream Health ダッシュボード:

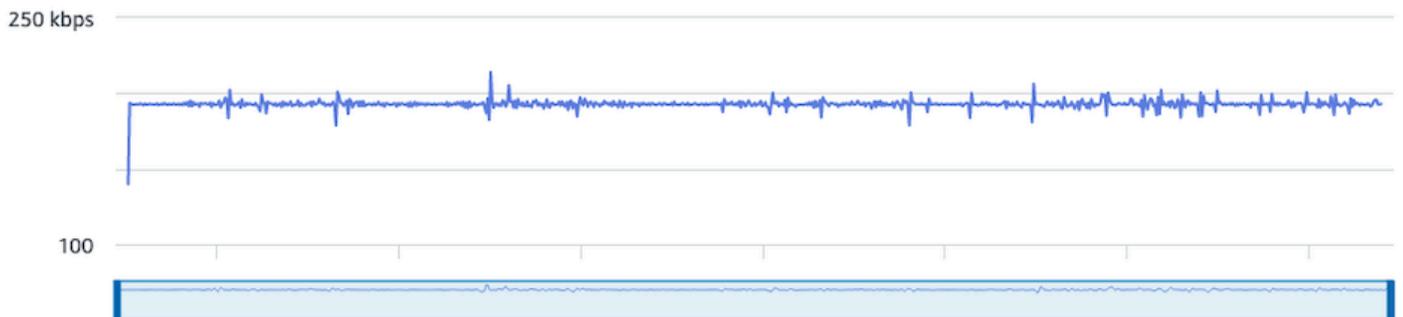
Video bitrate



Frame rate



Audio bitrate



CloudWatch:

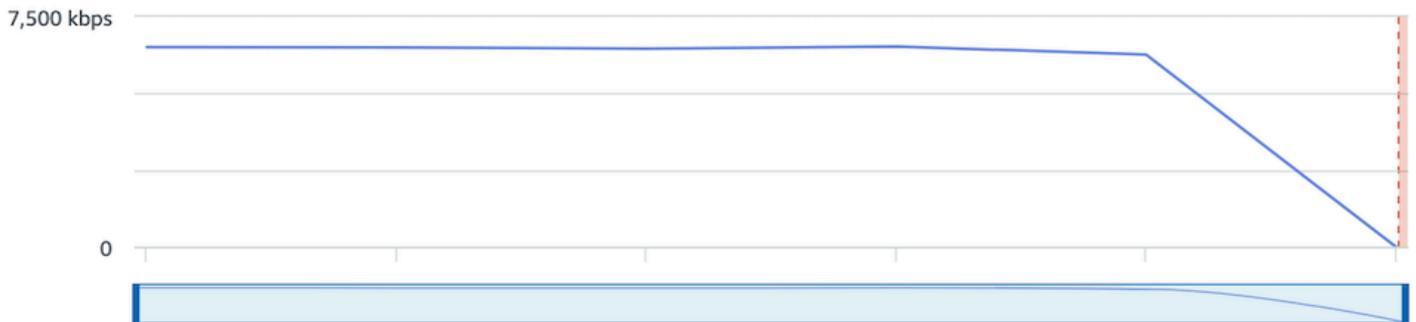


インターネットの切断

ブロードキャストデバイスでインターネットの問題が発生すると、IVS サーバーでは 30 秒間の期間が開始されて、この間に同じ接続が再確立されたかどうかを評価します。同じ接続が再確立されない場合、IVS サーバーはストリームセッションを終了します。一部のエンコーダーは、インターネット接続が失われた場合にブロードキャストセッションへの再接続を試みます。その場合、最初のストリームの終了後に新しいストリームセッションが開始される場合があります。

IVS Stream Health ダッシュボード:

Video bitrate



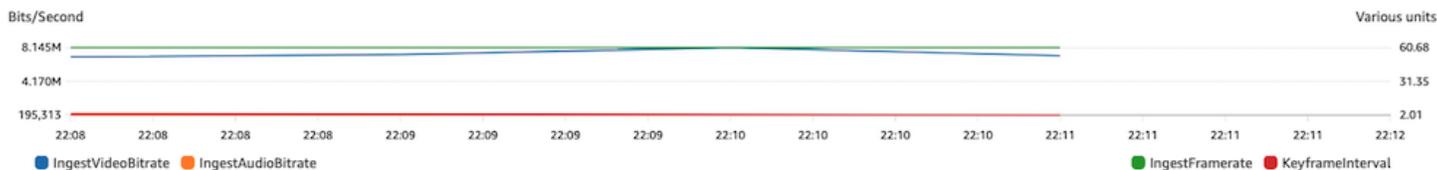
Frame rate



Audio bitrate



CloudWatch:



ストリーム再生

このセクションの情報のほとんどは、IVS Player SDK に固有のものであり、他のプレイヤーには適用されない場合があります。詳細については、「[Amazon IVS Player](#)」を参照してください。

トピック:

- [the section called “IVS プレイヤーの動作をデバッグするにはどうすればよいですか?”](#)
- [the section called “すべての視聴者の再生がフリーズ/停止したのはなぜですか?”](#)
- [the section called “IVS プレイヤーがバッファリングする原因は何ですか?”](#)

IVS プレイヤーの動作をデバッグするにはどうすればよいですか?

詳細なログ記録を有効にして IVS Player のデバッグを支援するには、`setLogLevel` プレイヤーメソッドを使用します。DEBUG 引数を使用するようにプレイヤーのログレベルを変更します。その後、IVS Player は、IVS Player で発生している状態とロジックに関する詳細なログ記録を生成します。

DEBUG ログを有効または無効にして、IVS Player を使用して迅速にテストするには、<https://debug.ivsdemos.com/> テストサイトを使用します。設定メニューを介して DEBUG ログが有効になっている場合は、ブラウザのコンソールビューでログを表示できます。

すべての視聴者の再生がフリーズ/停止したのはなぜですか?

コンテンツ内で同時にすべての視聴者の再生がフリーズまたは停止する場合は、アップストリームの動作が原因であると考えられます。多くの場合、根本原因はブロードキャストエンコーダーです。

[ストリームスタベーション](#)やブロードキャストエンコーダーの悪影響をもたらす動作は、すべての視聴者に同時に影響を及ぼす可能性があります。ブロードキャストエンコーディングが切断され、新しいストリームセッションが開始されると、すべての視聴者が同時にコンテンツを受信しなくなります。この動作を評価する場合は、「[Amazon IVS Low-Latency Streaming のモニタリング](#)」を使用してストリームセッションを評価することをお勧めします。

IVS プレイヤーがバッファリングする原因は何ですか？

ライブストリーミングの動画と音声の再生において、「バッファリング」とは、コンテンツが再生されることになっている時点よりも前に再生デバイスがコンテンツをダウンロードできないことを意味します。バッファリングは、コンテンツがランダムに停止および開始する（「途切れ」とも呼ばれます）、コンテンツが長時間にわたって停止する（「フリーズ」とも呼ばれます）、またはプレイヤーが BUFFERING 状態に入るなど、いくつかの態様で現れる可能性があります。

バッファリングを発生させる原因は多くありますが、主に次の 3 つのカテゴリに分類できます。

- 視聴者側のバッファリングは、多くの場合、単一の視聴者または少数の視聴者グループがバッファリングイベントの影響を受ける場合に発生します。これらのバッファリングイベントの根本原因は、ローカルネットワーク (LAN) または再生デバイスの問題から生じていることがよくあります。ローカルネットワークまたはデバイスの動作が遅いという問題の場合、アダプティブビットレート再生 (ABR) が有効になっていることを確認するか、手動でより低い品質を選択するか、他のプログラムやデバイスで使用されている帯域幅を減らすことで、バッファリングを解決できる場合があります。
- ネットワークレベルのバッファリング — ローカルネットワークと IVS ディストリビューションサーバーの間で問題が発生する可能性があります (ISP レベルとも呼ばれます)。ISP に関して完全な可視性を得ることができない場合があるため、ISP レベルで発生するバッファリング動作のトラブルシューティングが困難な場合があります。レイテンシーやネットワークの負荷 (ISP が全体的な着信/送信トラフィックを処理できないなど) の動作により、視聴者へのコンテンツの提供に遅延が生じる可能性があります。
- ブロードキャスト側のバッファリング — ライブストリームセッションのブロードキャスト側の問題は、視聴者のバッファリングに関する大規模な問題を引き起こす可能性があります。例えば、ブロードキャストデバイスが IVS へのデータ送信を停止した場合、IVS にはプレイヤーに配信するコンテンツがなく、コンテンツがダウンロードされていないときに IVS Player はバッファリング状態になります。多くの場合、ブロードキャスト側のバッファリングイベントにより、すべてではないにしても、ほとんどの視聴者が同時に影響を受けます。

Amazon S3 への自動録画

詳細については、「[Amazon S3 への自動録画](#)」を参照してください。

トピック:

- [the section called “一部の録画コンテンツが欠落しているのはなぜですか?”](#)

- [the section called “KMS-S3 暗号化は S3 への自動録画で使用できますか？”](#)

一部の録画コンテンツが欠落しているのはなぜですか？

録画されたコンテンツが欠落する理由はさまざまです。欠落しているコンテンツをトラブルシューティングするには、次のステップを実行することをお勧めします。

1. 必要な IVS チャンネルについて、S3 への自動録画が有効になっているようにします。
 - a. コンソール — 関連するチャンネルの詳細ページの [General configuration] (全般設定) セクションで、S3 への自動録画が Enabled であることを確認します。有効になっている場合は、録画設定をチェックして、ストレージと録画プレフィックスの両方が正しいことを確認します。
 - b. CLI — 目的の IVS チャンネル ARN で `get-channel` を実行して渡します。

```
aws ivs get-channel --arn "arn:aws:ivs:us-west-2:123456789012:channel/  
abcdABCDefgh"
```

`recordingConfigurationArn` が返されるかどうかを確認します。

2. 指定された S3 バケットで、特定のストリームセッションの記録コンテンツを確認します ([「S3 プレフィックス」](#)を参照)。録画されたセッションの S3 キープレフィックスは、Amazon EventBridge [Recording State Change イベント](#)にあります。注: [フラグメント化されたストリームをマージする](#)機能が有効になっている場合、一部のコンテンツが別の録画されたセッションになることがあります。
3. ストリーム全体の長さが 10 秒未満の場合、またはストリームのコンテンツが欠落している (ストリームスタベーションの発生など) 場合、何も生成されなかったために録画されたコンテンツが欠落している可能性があります。

KMS-S3 暗号化は S3 への自動録画で使用できますか？

Amazon S3 への IVS 自動録画機能は、[KMS-S3 暗号化](#) をサポートしていません。KMS-S3 暗号化を使用しようとする、録画開始は失敗し、[録画開始失敗 EventBridge イベント](#)が生成されます。推奨される回避策は、サポートされている [SSE-S3 暗号化を使用することです](#)。これは、Amazon S3 にアップロードされたすべてのオブジェクトでデフォルトで有効になっています。

その他のトピック

このセクションの質問は、他に分類できないトピックに関するものです。

トピック:

- [the section called “「pending verification」 \(検証待ち\) エラーは何を意味していますか?”](#)
- [the section called “IVS の使用コストを見積もることはできますか?”](#)

「pending verification」 (検証待ち) エラーは何を意味していますか？

IVS を使用すると、次のようなエラーが表示されることがあります: 「アカウントは検証待ちです。確認プロセスが完了するまでは、このアカウントでリクエストを実行できない可能性があります。質問がある場合は、AWS サポートに連絡してください。」

これは、IVS を使用する前に、使用している AWS アカウントを AWS で検証する必要があることを示しています。(アカウントで他の AWS のサービスを利用できる場合もありますが、IVS では強化された検証方法を採用しています)。

AWS アカウントを検証するには、AWS サポートセンター (<https://support.console.aws.amazon.com/support/home?#/>) から、受け取ったエラーメッセージを記載して AWS アカウントサポートまでお問い合わせください。

IVS の使用コストを見積もることはできますか？

ストリームセッションの前に IVS の使用に関する正確なコストを知ることはできませんが、大まかなコストの見積りツールは <https://ivs.rocks/calculator> でご利用いただけます。追加の料金情報は、<https://aws.amazon.com/ivs/pricing/> を参照してください。

望ましくないコンテンツと視聴者

悪意のあるユーザーが、プラットフォームで望ましくないコンテンツ (プロスポーツなど) を再ストリーミングしようとする可能性があります。この種のストリーミングは、ビジネスに価値を付加することなく、アプリケーションが提供するライブストリーミングビデオの量とそれに関連するコストを劇的に増加させる可能性があります。アクティブなストリームを停止するためのコントロールを提供することに加えて、Amazon IVS は、この種の動作を検出して防止するのに役立つリソースを提供します。

望ましくないコンテンツの検出

異常検出

特定の望ましくないコンテンツがストリーミングされているときに発生する、視聴率の異常な急上昇を検出してアラートを発することができます。(急上昇が発生したことが検出されたら、以下で説明するように、「[ストリームを停止して、ストリームキーをリセットする](#)」に記載されているステップを実行できます。)

Amazon CloudWatch では、視聴率が急増した場合など、特定の状況下でアラートを送信できるアラームを作成できます。Amazon IVS は、すべてのチャンネル CloudWatch の同時ビュー (CCV) メトリクスを Amazon に自動的にレポートするため、アラームを設定するだけで済みます。CCV に基づいて異常検出アラームを設定するには、次の手順に従います。

1. <https://console.aws.amazon.com/cloudwatch/> で Amazon CloudWatch コンソールを開きます。
2. 左側のナビゲーションバーで、[アラーム] を選択してから、[すべてのアラーム] を選択します。
3. ページの右上にある [アラームを作成] を選択します。
4. [メトリクスを選択] を選択します。メトリクスで IVS を選択し、次にすべての を選択し、 の横にあるチェックボックスをオンにします ConcurrentViews。
5. 右下にある [メトリクスを選択] を選択します。4 つのステップのアラーム作成ウィザードが開きます。
6. ウィザード: ステップ 1、[メトリクスと条件の指定] で、次の設定を指定します。
 - a. [統計] = [最大]
 - b. 期間 = 1 分
 - c. [しきい値タイプ] = [異常検出]
 - d. [同時再生数が次の場合は常時] = [バンドより大きい]

e. [異常検出のしきい値] = [3]

このしきい値は参考用の提案です。典型的なトラフィックパターンとニーズに応じて、別の値を選択することをお勧めします。メトリクスをより詳しくモニタリングするには、より低い値を使用します。アラームの数を減らすには、値を大きくします。

f. [次へ] を選択します。

7. ウィザード: ステップ 2、[設定アクション] で、既存の SNS トピックを選択するか、新しい SNS トピックを作成して、指定したアドレスに E メールを送信します。E メールを送信するトピックを作成するには、[新しいトピックを作成] を選択し、トピック名を指定して、メールアドレスを入力し、[トピックを作成] を選択します。[Select] (次へ) をクリックして次に進みます。
8. ウィザード: ステップ 3、[名前と説明を追加] で、アラームの名前とオプションの説明を追加し、[次へ] を選択します。
9. ウィザード: ステップ 4、[プレビューと作成] で、情報が正しいことを確認してから、[アラームを作成] を選択します。
10. アラームが作成されました。プロンプトが表示されたら、SNS サブスクリプションを確認するための指示に従います。

詳細については、以下を参照してください。

1. [Amazon IVS Low-Latency Streaming のモニタリング](#)
2. [異常検出に基づく CloudWatch アラームの作成](#)

カスタムコンテンツモデレーション

画像認識を介して望ましくないコンテンツを検出する、カスタムコンテンツモデレーションソリューションを詳しくご覧いただけます。Amazon IVS は、この種のソリューションで使用するサムネイル画像の生成を含め、[Amazon IVS ライブストリームを Amazon S3 に自動録画](#)する機能を提供します。

これらの検出手法と防止手法の導入を検討してください。

- [Amazon Rekognition を使用した Amazon IVS モデレーションデモ](#)では、S3 への IVS 自動録画と Amazon Rekognition を組み合わせてライブコンテンツをモデレートする方法を紹介しています。
- [Hive コンテンツモデレーションを Amazon IVS ビデオストリームに追加する](#)
- [AI/ML コンテンツモデレーションによるより安全なオンラインコミュニティの作成](#)は、IVS アプリケーション内での Amazon Rekognition 使用に関するブログ投稿です。

望ましくないコンテンツや視聴者の防止

ストリームを停止して、ストリームキーをリセットする

望ましくないコンテンツのストリーミングにチャンネルが使用されていることを検出した場合は、Amazon IVS コンソールを使用してストリームをシャットダウンできます。

1. [Amazon IVS コンソール](#)を開きます。([AWS マネジメントコンソール](#) から Amazon IVS コンソールにアクセスすることもできます。)
2. 必要に応じて、ナビゲーションバーから [リージョンを選択] ドロップダウンを使用して、チャンネルがホストされているリージョンを選択します。
3. 停止するストリームが実行されているチャンネルを選択します。
4. チャンネルページで、[ライブストリーム] セクションまで下方向に移動し、[ストリームを停止] を選択します。

ストリームを停止した後でも、ブロードキャスターはそのチャンネルでストリームを再開できます。これを防ぐには、ストリームキーをリセットします。これにより、ブロードキャスターが新しいストリームキーを取得せずにストリームを再開するのを防ぎます。ストリームキーをリセットするには:

- チャンネルページにいる間に、[ストリーム設定] セクションまで下方向に移動し、[ストリームキーをリセット] を選択します。

プログラムでストリームを停止し、ストリームキーをリセット (削除/作成) することもできます。「[Amazon IVS Low-Latency Streaming](#)」を参照してください。

アプリケーションがストリームキーを発行する方法によっては、新しいストリームキーが取得されないように、さらに対策を講じる必要がある場合があります。

プライベートチャンネルを使用する

多くの場合、サードパーティーのウェブサイトに再生 URL を埋め込むだけで、望ましくないコンテンツがプラットフォーム外の多数の視聴者にストリーミングされます。この種の動作を防ぐ最善のソリューションは、Amazon IVS プライベートチャンネルです。プライベートチャンネルを使用すると、有効な再生トークンを持つ視聴者のみが再生できるように制限できます。再生トークンは、再生アプリケーション内で視聴者を検証するために使用され、意図しないプラットフォームでの視聴を防ぐことができます。さらに、オリジンの強制を有効にすることができます。これにより、ドメインでホストされていないウェブサイトで視聴者がストリームを視聴するのを防ぐことができます。この保護を拡

張して、オリジンの厳格な強制も有効にすることで、一般的なストリーミングアプリケーションをカバーできます。

ユーザーに正式なアカウントの作成やログインを強制することなく、プライベートチャンネルと認証の保護を活用できることに注意してください。再生アプリケーションは、舞台裏で匿名でトークンを簡単に取得できます。オリジンの強制を引き続き利用できます。

プライベートチャンネルの詳細については、次を参照してください。

- 「IVS 低レイテンシーストリーミングユーザーガイド」の「[プライベートチャンネルの設定](#)」。そのドキュメント内で、オリジンの強制の詳細については、「[再生トークンの生成と署名](#)」を参照してください。
- [Amazon IVS を使用した承認されたライブストリーム再生用のプライベートチャンネルの作成](#) (ブログ記事)

再生制限ポリシーを使用する

[プライベートチャンネル](#) を使用しない場合でも、再生制限ポリシーを活用することで、同じ保護の一部を活用できます。これらのポリシーにより、パブリックチャンネルで GeoBlocking やオリジン強制などの機能を有効にできます。IVS コンソールまたは API を使用して再生制限ポリシーを作成し、そのポリシーの ARN をチャンネルにアタッチします。

再生制限ポリシーの詳細については、以下を参照してください。

- [IVS 低レイテンシーストリーミングの開始](#) — 望ましくないコンテンツや視聴者の防止に関する情報を参照してください。
- [IVS 低レイテンシーストリーミング API リファレンス](#) — 「再生制限ポリシーエンドポイントと PlaybackRestrictionPolicy オブジェクト」を参照してください。

コスト (低レイテンシーのストリーミング)

Amazon IVS ライブ動画のコストと、Amazon S3 への自動録画機能に関連するコストがあります。

ライブ動画

[Amazon IVS 料金モデル](#)には、動画の入力と出力に個別の料金が組み込まれています。

動画入力の料金は、使用するチャンネルタイプに応じて異なります。チャンネルタイプの詳細については、「IVS ストリーミング設定」の[チャンネルタイプ](#)を参照してください。

ユースケースに適したチャンネルタイプを選択するためのヘルプについては、コンソールの「選択のヘルプ」ツールを使用してください。

1. コンソールの [チャンネルの作成] ページで、[カスタム設定] を選択します。
2. [チャンネルタイプ] で [選択のヘルプ] を選択します。
3. レコメンデーションが提供されるまでプロンプトに従い、次に [レコメンデーションを選択する] を選択します。

動画出力では、視聴者に配信される動画に対して 1 時間あたりの料金が発生します。料金は、解像度と「請求リージョン」(動画の配信元)によって異なります。動画出力の料金は使用量に基づき階層化されており、無料利用枠が含まれています。

[IVS Cost Estimator](#) は便利なインタラクティブなコスト見積もりツールです。チャンネルタイプ、解像度、ストリーミング時間、視聴者の数、請求リージョンの値を入力します。コスト見積りの際は、次の経験則に注意します。

- 視聴者は参加したり退出したりし、平均して 50% のストリームが「配信」されています。Cost Estimator には、「視聴者の平均再生時間」のセクターが含まれています。デフォルトは 50% です。有料イベントは視聴者数が多くなると予想します。ただし、チケットを持つすべての視聴者が同時に視聴するとは限りません。
- 一部の視聴者は、ブロードキャストのソース解像度よりも低い解像度で視聴します。これは、特に高解像度のストリームの場合です。視聴者の一部は、より低コストな低解像度で視聴します。帯域幅、ネットワーク状態、ISP、ハードウェアなど、視聴者のさまざまな制約が理由です。
- タイミングは重要です。例えば、配信が学校、職場、休暇と重なる場合、オーディエンスのサイズに影響する可能性があります。

- ライブ以外のユーザーからライブオーディエンスを築くことは非常に難しいです。もちろん、例外もあります。外部のタレント（独自のフォロワーを持つインフルエンサーなど）を呼び込むと、オーディエンスのサイズが大きくなる可能性があります。

Amazon S3 への自動録画

Amazon S3 への自動録画機能の使用や S3 への書き込みには、Amazon IVS の料金はかかりません。Amazon S3 ストレージ、お客様に代わって Amazon IVS が行う S3 API 呼び出し、および保存された動画の視聴者に対する配信については、利用料金が発生します。

録画した動画の保存

お客様は IVS コンソールを使用して、S3 ストレージのニーズとコストの見積もりを生成できます。ユーザーがコンソールを使用してチャンネルの録画を設定する場合（チャンネルの作成時またはそれ以降）、推定されるデータ使用量が表示されます。このデータ使用量の推定値を、[S3 用の AWS 料金計算ツール](#)に入力して S3 ストレージとデータ移動の月額コストを見積もることができます。

コンソールで新しいチャンネルを作成するとき、または既存のチャンネルを編集するとき、[ストリームの記録と保存] の [自動録画の有効化] を選択します。ここでは、関連コストに関する情報が表示されます。

Record and store streams [Info](#)

Auto-record to S3 [Info](#)

For improved redundancy, always record locally via your streaming tool.

Enable automatic recording

Recording configuration

configuration-1 ▼



Create recording configuration

State

Active

Storage

s3-bucket-name [↗](#)

Recording prefix [Info](#)

s3://ivs-r2s3-ivsstoragebucket-1kem14abgbit8/ivs/v1/298083573632/<attached_channel_id>/

Recorded renditions

All renditions

Merge fragmented streams

Disabled

Thumbnail recording

At 60-second intervals

Thumbnail storage

Store thumbnails sequentially

Thumbnail resolution

Source (same resolution as input stream)



Associated costs

There are four cost components to consider when enabling record to S3: storage, request and data retrieval, data transfer, and data management. [Estimate data use.](#)

► Tags [Info](#)

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Cancel

Create channel

Estimate data use (データ使用量の推定)を選択して、データ使用計算ツールを表示します。

Estimate data use



Channel type

The channel type to use in estimations.

Standard



Average input bitrate

8.5

Mbps



Only use numbers between 0 and 8.5

Input resolution

1080p



Input framerate

60fps



Recording configuration

Choose an existing recording configuration



Recorded renditions

All renditions

Thumbnail recording

Record at an interval - 60s

Target thumbnail resolution

Source (same resolution as input stream)

Thumbnail storage

Store thumbnails sequentially

Merge fragmented streams

Disabled

画面に表示されているように、表示される推定値を [AWS 料金計算ツール](#) に入力して、S3 ストレージとデータ移動によって発生する月額費用の見積もりが計算できます。

録画した動画の提供

録画した動画を視聴者に提供するコストは、使用されている CDN によって異なります。例えば、Amazon CloudFront の [価格設定ページ](#) を参照してください。

リソースとサポート (低レイテンシーストリーミング)

リソース

<https://ivs.rocks/> は、公開済みコンテンツ (デモ、コードサンプル、ブログ投稿) を閲覧し、コストを見積もり、ライブデモを通じて Amazon IVS を体験するための専用サイトです。

「[Getting Started with Amazon Interactive Video Service Series' Articles](#)」 (Amazon Interactive Video Service シリーズの記事を用いて使用を開始) は、Amazon IVS の使用に関する初心者向けの一連の記事です。記事では、投稿に埋め込まれたインタラクティブなデモを使用して、IVS API のステップバイステップのウォークスルーを紹介しています。すべてのデモは、埋め込まれた CodePen を介して投稿から直接実行できます。今後もさまざまなトピックをカバーしていきます。

Amazon IVS に関するさまざまなトピックのブログ投稿が多数あります。

- [AWS のブログ](#) サイトで、ページの右側にある [Product or solution] (製品またはソリューション) > [Media Services] (メディアサービス) > [Amazon Interactive Video Service] を選択して、Amazon IVS で絞り込みます。
- DEV コミュニティサイトの [こちら](#) を参照してください。

デモ

デモ、コードサンプル、ブログ投稿については、<https://ivs.rocks/examples> を参照してください。

パートナーソリューション

Amazon IVS では、[Amazon パートナーネットワーク \(APN\)](#) のサードパーティプロバイダーと提携して、ライブストリーミングアプリケーションを強化するテクノロジーソリューションを提供しています。パートナーソリューションには以下のいくつかの種類があります。

パートナーのタイプ	提供するソリューション
分析	ライブストリーミング動画アプリケーションの運用上およびビジネス上の分析情報を提供します。これらの分析情報は視聴者のエンゲージメントを高め、機会を特定することで投資収益率を向上させるのに役立ちます。

パートナーのタイプ	提供するソリューション
インタラクティブ	ライブストリーミング動画アプリケーションの視聴者とのエンゲージメントを高めるのに役立ちます。
フェイスフィルターと背景フィルター	ライブストリームの視聴者に表示される、配信者の顔や背景表示を変更できます。

分析

[Bitmovin](#) の Analytics は、Amazon IVS Player 向けに構築された [analytics collector](#) を備えたフルマネージドサービスです。Analytics により、デバイス間の再生状態の追跡および監視、視聴者の人口統計の把握、再生体験の品質の監視を行い、視聴者に影響する問題をすばやく特定できます。

Bitmovin の Analytics は、すべてのチャンネルで収集された実用的なデータにより、視聴者、体験品質 (QoE)、主要なエラーに関する指標ダッシュボードを通じて、視聴者のエンゲージメントと定着率を高めるのに役立ちます。

30 のフィルターと分類を含む約 40 の指標にアクセスできます。また、Bitmovin の API とデータエクスポートを通じて、200 種類のディメンションとフィルターを利用できます。

Bitmovin Analytics を Amazon IVS Player SDK と統合するには、次の [Android](#) と [iOS](#) の使用開始ガイドを参照してください。

インタラクティブ

[LiveLike](#) は、すぐに使えるエンゲージメントプラットフォームを提供し、わずか数週間でオンラインユーザーエクスペリエンスを向上させることができます。登録、インタラクション、インプレッション、スポンサーシップを増やすことで、ユーザーあたりの平均収益を増やします。NASCAR のケーススタディの結果では、登録数が前年比で 70% 増加 (2022 年対 2021 年) しています。当社のソリューションを使用して、プラットフォーム上でインタラクティブで魅力的なエクスペリエンスを作成することで、解約率を減らし、定着率を高めます。LiveLike を Amazon IVS と統合するには、次のブログ記事「[A Quick Guide to LiveLike: How to Enhance Live Stream Interactivity](#)」を参照してください。

フェイスフィルターと背景フィルター

DeepAR は、デジタル製品チーム向けの AR インフラストラクチャを構築するテクノロジー企業です。スタートアップ企業から上場企業まで、あらゆる規模の企業が当社のソフトウェアを使用して、

世界中にいる何十億人ものユーザーに世界クラスの AR 体験を提供しています。DeepAR を Amazon IVS と統合するには、「[Amazon IVS の統合](#)」の DeepAR のページを参照してください。

[BytePlus Effects](#) は、AR エフェクト、ステッカー、フィルターの膨大なライブラリを組み合わせ、アプリ開発者が視聴者とのエンゲージメントを深めるために必要なすべてのツールを提供します。BytePlus を Amazon IVS と統合するには、ブログ記事「[How to improve user engagement with real-time AR effects using BytePlus Effects and Amazon IVS](#)」を参照してください。

Camera Kit は、パートナーがアプリケーションや Web サイトで Snap AR テクノロジーを活用できるようにする Snap AR の SDK です。Camera Kit を使用すると、企業は顧客に新たな次元のエクスペリエンスをもたらし、Snap の基盤となる AR テクノロジーを展開することができます。Snap の Camera Kit SDK を使用して Snap AR レンズを Amazon IVS と統合するには、ブログ記事「[Amazon IVS と Snap の Camera Kit AR SDK でクリエイターの表現力を引き出し、ライブストリーミング体験を向上させる](#)」を参照してください。

サポート

[AWS サポートセンター](#)では、AWS ソリューションをサポートするツールと専門知識へのアクセスを提供する一連のプランを用意しています。すべてのサポートプランで、24 時間年中無休のカスタマーサービスをご利用いただけます。AWS 環境の計画、デプロイ、最適化のために技術サポートや追加のリソースが必要な場合は、お客様の AWS ユースケースに合ったサポートプランを選択してください。

[AWS プレミアムサポート](#)は、1 対 1 での迅速な対応を行うサポート窓口であり、AWS でのアプリケーションの構築と運用を支援します。

[AWS re:Post](#) は、Amazon IVS に関連する技術的な質問についてディスカッションするコミュニティベースの開発者向け Q&A サイトです。

[お問い合わせ](#) - 請求やアカウントに関する非技術的なお問い合わせ用のリンクがあります。技術的な質問の場合は、上記の AWS re:Post または AWS プレミアムサポートをご利用ください。

用語集

「[AWS 用語集](#)」も参照してください。下の表では、LL は IVS 低レイテンシーストリーミング、RT、IVS リアルタイムストリーミングを表します。

言葉	説明	LL	RT	チャット
AAC	高度なオーディオコーディング。AAC は、非可逆デジタルオーディオ 圧縮 用のオーディオコーディング規格です。MP3 形式の後継として設計された AAC は、通常、同じビットレートで MP3 よりも高い音質を実現します。AAC は MPEG-2 および MPEG-4 の仕様の一部として ISO と IEC によって標準化されています。	✓	✓	
アダプティブビットレートのストリーミング	アダプティブビットレート (ABR) ストリーミングにより、IVS プレーヤーは接続品質が低下した場合は低い ビットレート に切り替え、接続品質が向上した場合は高いビットレートに戻すことができます。	✓		
アダプティブストリーミング	「 サイマルキャストによるレイヤードエンコーディング 」を参照してください。		✓	
管理ユーザー。	AWS アカウントで利用可能なリソースとサービスへの管理アクセス権を持つ AWS ユーザー。 「AWS セットアップユーザーガイド」の「 用語 」を参照してください。	✓	✓	✓
ARN	AWS リソースに固有の識別子である Amazon リソースネーム 。具体的な ARN 形式は、リソースの種類によって異なります。IVS リソースで使用される ARN 形式については、「サービス認証リファレンス」を参照してください。	✓	✓	✓

言葉	説明	LL	RT	チャット
アスペクト比	フレーム幅とフレーム高さの比率について説明します。たとえば、16:9 はフル HD または 1080p の 解像度 に対応するアスペクト比です。	✓	✓	
オーディオモード	さまざまなタイプのモバイルデバイスユーザーや使用する機器に合わせて最適化された、プリセットまたはカスタムのオーディオ設定。「 IVS Broadcast SDK: モバイルオーディオモード (リアルタイムストリーミング) 」を参照してください。		✓	
AVC、H.264、MPEG-4 Part 10	アドバンストビデオコーディング (H.264 または MPEG-4 Part 10 と呼ばれる) は、非可逆デジタルビデオ 圧縮 用のビデオ圧縮規格です。	✓	✓	
背景置換	ライブストリームのクリエイターが背景を変更できるようにする カメラフィルター の一種。「IVS Broadcast SDK: サードパーティーのカメラフィルター (リアルタイムストリーミング)」の「 背景の置換 」を参照してください。		✓	
ビットレート	1 秒あたりに送受信されるビット数のストリーミングメトリック。	✓	✓	
ブロードキャスト、配信者	ストリーム 、 ストリーマー のためのその他の用語。	✓		
バッファリング	コンテンツが再生されることになっている時点よりも前に再生デバイスがコンテンツをダウンロードできない場合に発生する状態。バッファリングは、コンテンツがランダムに停止および開始する (「途切れ」とも呼ばれます)、コンテンツが長時間にわたって停止する (「フリーズ」とも呼ばれます)、または IVS プレイヤーが再生を一時停止するなど、いくつかの態様で現れる可能性があります。	✓	✓	

言葉	説明	LL	RT	チャット
バイト範囲プレイリスト	<p>標準の HLS プレイリスト よりも詳細なプレイリスト。標準 HLS プレイリストは 10 秒のメディアファイルで構成されています。バイト範囲のプレイリストでは、セグメント再生時間は ストリーム に設定された キーフレーム間隔 と同じです。</p> <p>バイトレンジプレイリストは、S3 バケット に自動記録されたブロードキャストでのみ使用できます。HLS プレイリスト に加えて作成されます。</p> <p>「Amazon S3 への自動レコーディング (低レイテンシーストリーミング)」の「バイトレンジプレイリスト」を参照してください。</p>	✓		
CBR	<p>コンスタントビットレートとは、ブロードキャスト中に起こる事象に関わらず、動画の再生中ずっと一定のビットレートを維持するエンコーダー用のレート制御方法です。アクション中の小康状態は希望のビットレートになるようにパディングされ、ピークはターゲットビットレートに合うようにエンコーディングの品質を調整することで量子化できます。VBR ではなく CBR を使用することを強くお勧めします。</p>	✓	✓	
CDN	<p>コンテンツ配信ネットワーク (Content Delivery Network または Content Distribution Network) は、ストリーミングビデオなどのコンテンツをユーザーのいる場所に近づけることで配信を最適化する、地理的に分散したソリューションです。</p>	✓		

言葉	説明	LL	RT	チャット
チャンネル	インジェストサーバー 、 ストリームキー 、 再生 URL 、録画オプションなど、ストリーミングの設定を保存する IVS リソース。ストリーマーは、チャンネルに関連付けられたストリームキーを使用してブロードキャストを開始します。すべてのメトリクスとブロードキャスト中に生成される イベント は、チャンネルリソースに関連付けられています。	✓		
チャンネルタイプ	チャンネル の許容 解像度 と フレームレート を決定します。「IVS 低レイテンシーストリーミング API リファレンス」の「 チャンネルタイプ 」を参照してください。	✓		
チャットのログ記録	ログ記録設定を チャットルーム に関連付けることで有効にできる詳細オプション。			✓
チャットルーム	メッセージレビューハンドラー や チャットロギング などのオプション機能を含む、チャットセッションの設定を保存する IVS リソース。「IVS Chat の開始方法」の「 ステップ 2: チャットルームを作成する 」を参照してください。			✓
クライアントサイドコンポジション	ホスト デバイスを使用してステージ参加者からのオーディオストリームとビデオストリームをミックスし、それらをコンポジットストリームとして IVS チャンネル に送信します。これにより、クライアントリソースの使用率が高くなり、 ステージ や ホスト の問題が視聴者に影響を与えるリスクは高まりますが、 コンポジション の外観をより細かく制御できます。 「 サーバーサイドコンポジション 」も参照してください。	✓	✓	

言葉	説明	LL	RT	チャット
CloudFront	Amazon が提供する CDN サービス。	✓		
CloudTrail	AWS や外部ソースからのイベントやアカウントアクティビティを収集、監視、分析、保持するための AWS サービス。 「AWS での IVS API コールのログ記録 CloudTrail」 を参照してください。	✓	✓	✓
CloudWatch	アプリケーションの監視、パフォーマンスの変化への対応、リソース使用の最適化、および運用状況に関するインサイトの提供を行う AWS サービス。CloudWatch を使用して IVS メトリクスをモニタリングできます。 「IVS リアルタイムストリーミングのモニタリング」 および 「IVS 低レイテンシーストリーミングのモニタリング」 を参照してください。	✓	✓	✓
コンポジション	複数のソースからのオーディオストリームとビデオストリームを 1 つのストリームにまとめるプロセス。	✓	✓	
コンポジションパイプライン	複数のストリームを結合し、結果のストリームをエンコードするために必要な一連の処理ステップ。	✓	✓	
圧縮	元の表示よりも少ないビット数で情報をエンコードします。いずれの圧縮も、可逆圧縮または非可逆圧縮です。可逆圧縮は、統計上の冗長性を特定して排除することでビット数を削減します。可逆圧縮では情報が失われることはありません。非可逆圧縮は、不要な情報や重要度の低い情報を削除することでビット数を削減します。	✓	✓	

言葉	説明	LL	RT	チャット
コントロールプレーン	チャンネル 、 ステージ 、 チャットルーム などの IVS リソースに関する情報を保存し、これらのリソースを作成および管理するためのインターフェースを提供します。リージョナルであり、AWS リージョン に基づきます。	✓	✓	✓
CORS	Cross-Origin Resource Sharing は、特定のドメインにロードされたクライアントウェブアプリケーションが異なるドメイン内 S3 バケット などのリソースと通信する方法を定義します。アクセスはヘッダー、HTTP メソッド、オリジンドメインに基づいて設定できます。「Amazon Simple Storage Service ユーザーガイド」の「 クロスオリジンリソース共有 (CORS) の使用 – Amazon Simple Storage Service 」を参照してください。	✓		
カスタム画像ソース	IVS Broadcast SDK が提供するインターフェース。プリセットカメラに限定されず、アプリケーションが独自の画像入力を行えるようにします。	✓	✓	
データプレーン	データを 取り込み から出力まで伝送するインフラストラクチャ。 コントロールプレーン で管理される設定に基づいて動作し、AWS リージョンに限定されません。	✓	✓	✓
エンコーダ。エンコーディングします	動画やオーディオコンテンツをストリーミングに適したデジタル形式に変換するプロセス。エンコーディングはハードウェアベースでもソフトウェアベースでもかまいません。	✓	✓	

言葉	説明	LL	RT	チャット
イベント	IVS がモニタリングサービスに発行する AmazonEventBridge 自動通知。イベントは、 ステージ や コンポジションパイプライン などのストリーミングリソースの状態や状態の変化を表します。 「Amazon EventBridge と IVS 低レイテンシーストリーミングの使用」 および 「Amazon EventBridge と IVS リアルタイムストリーミングの使用」 を参照してください。	✓	✓	✓
FFmpeg	動画やオーディオのファイルやストリームを処理するためのライブラリとプログラム群で構成される、無料でオープンソースのソフトウェアプロジェクト。 FFmpeg は、オーディオと動画を録画、変換、ストリーミングするためのクロスプラットフォームソリューションを提供します。	✓		
断片化されたストリーム	ブロードキャストが切断され、 チャンネル の録画設定で指定された時間内に再接続されるときに作成されます。生成された複数のストリームは、単一のブロードキャストと見なされ、マージされ単一の録画ストリームになります。「Amazon S3 への自動記録(低レイテンシーストリーミング)」の 「フラグメント化されたストリームのマージ」 を参照してください。	✓		
フレームレート	1 秒あたりに送受信される動画フレーム数のストリーミングメトリック。	✓	✓	
HLS	HTTP ライブストリーミング (HLS) は、IVS ストリームを視聴者に配信するために使用される HTTP ベースの アダプティブビットレートストリーミング 通信プロトコルです。	✓		

言葉	説明	LL	RT	チャット
HLS のプレイリスト	ストリームを構成するメディアセグメントのリスト。標準 HLS プレイリストは 10 秒のメディアファイルで構成されています。HLS は、より詳細な バイト範囲のプレイリスト をサポートしています。	✓		
ホスト	ビデオやオーディオをステージに送信するリアルタイムのイベント 参加者 。		✓	
IAM	Identity and Access Management は、ユーザーが ID を管理し、IVS を含む AWS のサービスとリソースへのアクセスを安全に管理できるようにする AWS サービスです。	✓	✓	✓
取り込み	IVS は、ホストまたはブロードキャストから動画ストリームを受信して処理または視聴者や他の参加者に配信するためのプロセスです。	✓	✓	
取り込みサーバー	ビデオストリームを受信してトランスコーディングシステムに配信します。トランスコーディングシステムでは、ストリームが トランスミックス されるか、 HLS にトランスコーディング され、視聴者に配信されます。 インジェストサーバーは、取り込みプロトコル (RTMP 、 RTMPS) とともに チャンネル のストリームを受信する特定の IVS コンポーネントです。チャンネルの作成については、「 IVS 低レイテンシーストリーミングの開始 」を参照してください。		✓	
インターレースビデオ	後続のフレームの奇数行または偶数行のみを送信して表示し、余分な帯域幅を消費せずに、体感 フレームレート を倍増させます。ビデオ品質の問題から、インターレースビデオの使用はお勧めしません。	✓	✓	

言葉	説明	LL	RT	チャット
JSON	JavaScript Object Notation は、人間が読み取れるテキストを使用して、属性と値のペア、配列データ型、またはその他の直列化可能な値で構成されるデータオブジェクトを送信するオープンスタンダードのファイル形式です。	✓	✓	✓
キーフレーム、デルタフレーム、キーフレーム間隔	キーフレーム (イントラコードまたは i フレームとも呼ばれます) は、動画内の画像のフルフレームです。後続のフレームであるデルタフレーム (予測フレームまたは P フレームとも呼ばれます) には、変更された情報のみが含まれます。キーフレームは、エンコーダーで定義されているキーフレーム間隔に応じて、 ストリーム 内に複数回表示されます。	✓	✓	
Lambda	サーバーインフラストラクチャをプロビジョニングせずにコード (Lambda 関数と呼ばれる) を実行するための AWS サービス。Lambda 関数は、イベントや呼び出しリクエストに応答して実行することも、スケジュールに基づいて実行することもできます。たとえば、IVS Chat は Lambda 関数を使用して チャットルームのメッセージレビュー を有効にします。	✓	✓	✓
レイテンシー、glass-to-glassレイテンシー	<p>データ転送の遅延。IVS はレイテンシー範囲を次のように定義しています。</p> <ul style="list-style-type: none"> 低レイテンシー: 3 秒未満 リアルタイムレイテンシー: 300 ms 未満 <p>Glass-to-glass レイテンシーとは、カメラがライブストリームをキャプチャしてから視聴者の画面に表示されるまでの遅延を指します。</p>	✓	✓	

言葉	説明	LL	RT	チャット
サイマルキャストによるレイヤードエンコーディング。	品質レベルの異なる複数のビデオストリームを同時にエンコードして公開できます。「リアルタイムストリーミングによる最適化」の「 アダプティブストリーミング: サイマルキャストによるレイヤードエンコーディング 」。		✓	
メッセージレビューハンドラ	IVS Chat の顧客に、 チャットルーム に配信される前にユーザーチャットメッセージを自動的に確認/フィルタリングする機能を付与します。 Lambda 関数をチャットルームに関連付けることで有効になります。「チャットメッセージレビューハンドラ」の「 Lambda 関数の作成 」を参照してください。			✓
ミキサー	IVS Mobile Broadcast SDK の機能の 1 つとして、複数のオーディオおよびビデオソースを受け取り、単一の出力を生成します。カメラ、マイク、スクリーンキャプチャ、アプリケーションで生成されたオーディオと動画などのソースを表す画面上のビデオとオーディオ要素の管理をサポートします。その後、出力を IVS にストリーミングできます。「IVS Broadcast SDK: Mixer ガイド (低レイテンシーストリーミング)」の「 ミキシング用のブロードキャストセッションの設定 」を参照してください。	✓		

言葉	説明	LL	RT	チャット
マルチホストストリーミング	<p>複数のホストからのストリームを1つのストリームにまとめます。これは、クライアント側またはサーバー側のコンポジションを使用して実現できます。</p> <p>マルチホストストリーミングでは、視聴者をステージに招待して質疑応答、ホスト同士の競争、ビデオチャット、大勢の視聴者の前でホスト同士が会話するなどのシナリオが可能になります。</p>		✓	
マルチバリエーションプレイリスト	ブロードキャストで利用できるすべての バリエーションストリーム のインデックス。	✓		
OAC	SS3 バケット へのアクセスを制限し、録画されたストリームなどのコンテンツを CloudFront CDN 経由でのみ提供できるようにするメカニズムであるオリジンアクセスコントロール。	✓		
OBS	オープンブロードキャストソフトウェア (OBS) – 動画の録画とライブストリーミング用のオープンソースの無料ソフトウェア。 OBS はデスクトップパブリッシング用の (IVS ブロードキャスト SDK に代わる) 代替手段を提供します。シーントランジション、オーディオミキシング、オーバーレイグラフィックなどの高度な制作機能を備えているため、OBS に精通している上級ストリーマーには OBS が好ましいかもしれません。	✓	✓	
Participant	ホスト または 視聴者 としてステージに接続したリアルタイムのユーザー。		✓	
参加者トークン	イベント 参加者 が ステージ に参加すると、その参加者をリアルタイムで認証します。参加者トークンは、参加者がステージに動画を送信できるかどうかを制御します。		✓	

言葉	説明	LL	RT	チャット
再生トークン、再生キーペア	<p>顧客がプライベートチャンネルでの動画再生を制限できるようにする認証メカニズム。再生トークンは、再生 キーペアから生成されます。</p> <p>再生キーペアは、再生用の視聴者認証トークンに署名して検証するために使用されるパブリックキーとプライベートキーのペアです。「プライベートチャンネルの設定」の「再生キーの作成またはインポート」と、「IVS 低レイテンシー API リファレンス」の「再生キーペアのエンドポイント」を参照してください。</p>	✓		
再生 URL	<p>視聴者が特定のチャンネルの再生を開始するために使用するアドレスを識別します。このアドレスはグローバルに使用できます。IVS は、IVS グローバルコンテンツ配信ネットワーク上で最適な場所を自動的に選択し、各視聴者に動画を配信します。チャンネルの作成については、「IVS 低レイテンシーストリーミングの開始」を参照してください。</p>	✓		
¥プライベートチャンネル	<p>再生トークンに基づく認証メカニズムを使用して、お客様がストリームへのアクセスを制限できるようにします。「プライベートチャンネルの設定」の「プライベートチャンネルのワークフロー」を参照してください。</p>	✓		
プログレッシブビデオ	<p>各フレームのすべての行を順番に送信して表示します。ブロードキャストのすべての段階でプログレッシブビデオを使用することをお勧めします。</p>	✓	✓	

言葉	説明	LL	RT	チャット
クォータ	AWS アカウントの IVS サービスリソースまたはオペレーションの最大数。つまり、これらの制限は、特に断りのない限り、AWS のアカウントごとに適用されます。すべてのクォータはリージョンごとに適用されます。「AWS 一般リファレンスガイド」の「 Amazon インタラクティブビデオサービスのエンドポイントとクォータ 」を参照してください。	✓	✓	✓
リージョン	<p>リージョンを使用すると、特定の地域に物理的に存在する AWS のサービスにアクセスすることができます。リージョンでは耐障害性や安定性が提供され、レイテンシーを低減することもできます。これにより、リージョンの障害の影響を受けずに利用できる冗長リソースを作成できます。</p> <p>ほとんどの AWS サービスのリクエストは特定の地域に関するものです。あるリージョンで作成したリソースは、AWS サービスで提供されるレプリケーション機能を明示的に使用しないかぎり、他のリージョンに存在することはありません。たとえば、Amazon S3 はクロスリージョンのレプリケーションをサポートしています。IAM などの一部のサービスには、リージョン間のリソースがありません。</p>	✓	✓	✓
解決方法	1 つの動画フレームのピクセル数を示します。たとえば、フル HD または 1080p は 1920 x 1080 ピクセルのフレームを定義します。	✓	✓	
ルートユーザー	AWS アカウントの所有者。ルートユーザーは、AWS アカウントのすべての AWS サービスとリソースへの完全なアクセス権を持ちます。	✓	✓	✓

言葉	説明	LL	RT	チャット
RTMP、RTMPS	Real-Time Messaging Protocol。ネットワーク上でオーディオ、動画、データを送信するための業界標準。RTMPS は、Transport Layer Security (TLS/SSL) 接続上で実行される RTMP の安全なバージョンです。	✓	✓	
S3 バケット	Amazon S3 に格納されたオブジェクトのコレクション。アクセスやレプリケーションを含む多くのポリシーがバケットレベルで定義され、バケット内のすべてのオブジェクトに適用されます。たとえば、IVS ブロードキャストは S3 バケット内の複数のオブジェクトとして保存されます。	✓		
SDK	ソフトウェア開発キットは、IVS を使用してアプリケーションを構築する開発者向けのライブラリ集です。	✓	✓	✓
自撮りセグメンテーション	カメラ画像を入力として受け取り、画像の各ピクセルがフォアグラウンドかバックグラウンドかを示す信頼度スコアを提供するマスクを返す、お客様固有のソリューションを使用して、ライブストリームのバックグラウンドを置き換えることができます。「IVS Broadcast SDK: サードパーティーのカメラフィルター (リアルタイムストリーミング)」の「 背景の置換 」を参照してください。		✓	
セマンティックバージョンニング	Major.Minor.Patch 形式のバージョンフォーマット。API に影響しないバグ修正ではパッチバージョンが上がり、下位互換性のある API を追加または変更するとマイナーバージョンが上がり、下位互換性のない API の変更ではメジャーバージョンが上がります。	✓	✓	✓

言葉	説明	LL	RT	チャット
サーバーサイドコンポジション	<p>IVS サーバーを使用してステージ参加者全員からの音声と動画を合成し、IVS チャンネル に送信し、より多くの視聴者や S3 バケット に保存します。サーバーサイドコンポジションにより、クライアントの負荷が軽減され、ブロードキャストの耐性が向上し、帯域幅をより効率的に使用できるようになります。</p> <p>「クライアント側コンポジション」も参照してください。</p>		✓	
Service Quotas	<p>AWS は、多くの AWS サービスの クォータ を 1 か所から管理できるサービスです。クォータ値を確認できるだけでなく、Service Quotas コンソールからクォータの引き上げをリクエストすることもできます。</p>	✓	✓	✓
サービスリンクロール	<p>AWS サービスに直接リンクされた一意のタイプの IAM ロールです。サービスにリンクされたロールは、IVS で自動作成され、サービスがユーザーの代わりに、S3 バケット へのアクセスなど、その他の AWS サービスを呼び出すために必要なすべてのアクセス権限が付与されます。「IVS Security」の IVS の「サービスにリンクされたロールの使用」を参照してください。</p>	✓		
ステージ	<p>リアルタイムのイベント参加者がリアルタイムでビデオを送受信できる仮想スペースを提示する IVS リソース。「IVS リアルタイムストリーミング入門」の「ステージの作成」を参照してください。</p>		✓	

言葉	説明	LL	RT	チャット
ステージセッション	最初の参加者が ステージ に参加したときに始まり、最後の参加者がステージへの公開を停止した数分後に終了します。長期間有効なステージでは、その存続期間中に複数のセッションが発生する場合があります。		✓	
ストリーム	ソースから宛先に継続的に送信される動画またはオーディオコンテンツを表すデータ。	✓	✓	
ストリームキー	チャンネル 作成時に IVS によって割り当てられる識別子で、チャンネルへのストリーミングの認証に使用されます。ストリームキーを使うとすべてのユーザーがチャンネルにストリーミングできるため、ストリームキーは機密情報として扱ってください。「 IVS 低レイテンシーストリーミングを開始する 」を参照してください。	✓		
ストリームスタベーション	IVS へのストリーム配信の遅延または停止。IVS が、エンコーディングデバイスがアダプタイズした想定されたビット数を、一定期間にわたって取り込まなかった場合に発生します。ストリーム不足が発生すると、ストリームスタベーション イベント が発生します。 視聴者の観点から見ると、ストリームスタベーションは、動画における遅延、バッファリング、フリーズとして現れる場合があります。ストリームスタベーションは、ストリーム不足の原因となった特定の状況に応じて、短い (5 秒未満) 場合もあれば、長い (数分) 場合もあります。「 トラブルシューティング FAQ 」の「 ストリームスタベーションとは 」を参照してください。	✓	✓	
ストリーマー	IVS にビデオまたはオーディオ ストリーム を送信するユーザーまたはデバイス。	✓	✓	

言葉	説明	LL	RT	チャット
サブスクライバー	ホストの動画やオーディオを受信するリアルタイムのイベント参加者。「 IVS リアルタイムストリーミングとは 」を参照してください。		✓	
タグ	AWS リソースに割り当てるメタデータラベル。タグを使用すると、AWS リソースの識別や整理ができます。 IVS ドキュメントのランディングページ では、IVS API ドキュメント (リアルタイムストリーミング、低レイテンシーストリーミング、チャット) の「タグ付け」を参照してください。	✓	✓	✓
サードパーティーのカメラフィルター	IVS Broadcast SDK と統合できるソフトウェアコンポーネント。これにより、アプリケーションは画像を カスタム画像ソース として Broadcast SDK に送信する前に処理できます。サードパーティーのカメラフィルターは、カメラからの画像を処理したり、フィルター効果を適用したりできます。	✓	✓	
サムネイル	ストリームから取得した縮小サイズの画像。デフォルトでは、サムネイルは 60 秒ごとに生成されますが、より短い間隔を設定することもできます。サムネイルの解像度は チャンネルタイプ によって異なります。「Amazon S3 への自動記録 (低レイテンシーストリーミング)」の「 コンテンツの記録 」を参照してください。	✓		

言葉	説明	LL	RT	チャット
時間指定メタデータ	<p>ストリーム内の特定のタイムスタンプに関連付けられたメタデータ。IVS API を使用してプログラムで追加でき、特定のフレームに関連付けられます。これにより、すべての視聴者は、ストリームに対してメタデータを同じ時点で受信できます。</p> <p>時限メタデータを使用して、スポーツイベント中にチームの統計を更新するなど、クライアント側でアクションをトリガーできます。「動画ストリーム内にメタデータを埋め込む」を参照してください。</p>	✓		
トランスコーディング	<p>動画とオーディオを、あるフォーマットから別のフォーマットに変換します。入力ストリームが複数のビットレートと解像度が異なるフォーマットにコード変換され、各種の再生デバイスとネットワーク条件をサポートします。</p>	✓	✓	
トランスマックス	<p>ビデオストリームの再エンコードを行わずに、取り込んだストリームを IVS に簡単に再パッケージ化できます。トランスマックスは、「transcode multiplexing」(トランスコード多重化) の略称で、オーディオおよび/または動画ファイルを、元のストリームの一部またはすべてを保持しながら、フォーマット変更するプロセスです。トランスマキシングは、ファイルの内容を変更せずに別のコンテナ形式に変換します。トランスコーディングとは区別されます。</p>	✓	✓	

言葉	説明	LL	RT	チャット
バリエーションストリーム	<p>同じブロードキャストを複数の異なる品質レベルでエンコードしたものです。各バリエーションストリームは個別の HLS プレイリスト としてエンコードされます。利用可能なバリエーションストリームのインデックスは、マルチバリエーションプレイリスト と呼ばれます。</p> <p>IVS プレーヤーは IVS からマルチバリエーションプレイリストを受信すると、再生中にバリエーションストリームの中から選択でき、ネットワークの状況の変化に応じてシームレスに切り替わります。</p>	✓		
VBR	<p>可変ビットレート。必要なディテールのレベルに応じて再生中に変化するダイナミックビットレートを使用するエンコーダのレート制御方法です。動画品質の問題から VBR は使用しないことを強くお勧めします。代わりに CBR を使用してください。</p>	✓	✓	

言葉	説明	LL	RT	チャット
ビュー	<p>アクティブに動画をダウンロードまたは再生している固有の視聴セッション。視聴回数は同時視聴クォータの基準です。</p> <p>視聴セッションの動画再生開始により、視聴が始まります。視聴セッションが動画の再生を停止すると、視聴が終了します。再生は視聴率の唯一の指標です。オーディオレベル、ブラウザのタブフォーカス、動画の品質などのエンゲージメントヒューリスティクスは考慮されません。IVS は視聴回数をカウントする際、個々の視聴者の正当性を考慮せず、また、1 台のマシンに複数の動画プレーヤーがあるなど、ローカライズされた視聴者の重複排除をしません。「Service Quotas (低レイテンシーストリーミング)」の「その他のクォータ」を参照してください。</p>	✓		
表示者	IVS から ストリーム を受信しているユーザ。	✓		
WebRTC	<p>ウェブリアルタイムコミュニケーションは、ウェブブラウザとモバイルアプリケーションにリアルタイムのコミュニケーションを提供するオープンソースプロジェクトです。これにより、直接通信が可能になり、オーディオとビデオ peer-to-peer の通信がウェブページ内で機能するため、プラグインをインストールしたり、ネイティブアプリケーションをダウンロードしたりする必要がありません。</p> <p>WebRTC の背後にあるテクノロジーはオープンなウェブ標準として実装され、すべての主要なブラウザで通常の JavaScript APIs として、または Android や iOS などのネイティブクライアントのライブラリとして使用できます。</p>	✓	✓	

言葉	説明	LL	RT	チャット
WHIP	<p>WebRTC - HTTP Ingestion Protocol。ストリーミングサービスや CDN へのコンテンツの WebRTC ベースの取り込み を可能にする HTTP ベースのプロトコルです。 CDNs WHIP は、WebRTC 取り込みを標準化するために開発された IETF ドラフトです。</p> <p>WHIP は OBS などのソフトウェアとの互換性を可能にし、デスクトップ公開用の代替 (IVS Broadcast SDK へ) を提供します。OBS に精通している、より高度なストリーマーは、シーンの移行、オーディオミキシング、オーバーレイグラフィックスなどの高度な制作機能よりも、OBS を優先する場合があります。</p> <p>WHIP は、IVS Broadcast SDK の使用が実行可能でない場合や推奨されない場合にも有益です。例えば、ハードウェアエンコーダーを含むセットアップでは、IVS Broadcast SDK はオプションではない場合があります。ただし、エンコーダーが WHIP をサポートしている場合でも、エンコーダーから IVS に直接公開できます。</p> <p>「OBS と WHIP サポート」を参照してください。</p>		✓	
WSS	<p>WebSocket 安全なプロトコルで、暗号化された TLS 接続 WebSockets を介して を確立します。IVS Chat エンドポイントへの接続に使用されています。「IVS Chat の開始方法」の 「ステップ 4: 最初のメッセージの送受信」 を参照してください。</p>			✓

ドキュメント履歴 (低レイテンシーストリーミング)

低レイテンシーストリーミングユーザーガイドの変更

変更	説明	日付
Service Quotas	「API コールレート クォータ」テーブルに、 StartViewerSession Revocation とを追加しま したBatchStartViewerSe ssionRevocation 。(こ れらは新しいエンドポイン トではありませんが、テー ブルにありませんでした)。 これらはエンドポイントと同 じテーブル部分にあります playback-key-pair 。エンドポ イントタイプは「プライベ ートチャンネル」です。	2024 年 2 月 5 日
Broadcast SDK: Android 1.14.1、iOS 1.14.1、Web 1.8.0	Broadcast low-latency- streaming SDK ガイド: Androidhttps://docs.aws. amazon.com/ivs/latest/Lo wLatencyUserGuide/broadcas t-ios.html 、iOS、および Web で、新しいリリースのバー ジョン番号とアーティファ クトのリンクが更新されま した。 Amazon IVS ドキュメ ントのランディングページ で は、Broadcast SDK リファレ ンスのリンクが更新され、新 しいバージョンを表示するよ うになりました。このリリー	2024 年 2 月 1 日

スについては、「[Amazon IVS リリースノート](#)」も参照してください。

Android ガイドに、新しい既知の問題 (ビデオサイズが 176 x 176 未満) を追加しました。

トークンレス再生の制限

このリリースでは、再生承認の外部でオリジンの強制とジオフェンシングが可能になります。いくつかの低レイテンシーストリーミングドキュメントが変更されました。

2024 年 1 月 31 日

- [開始方法](#) - 「ステップ 4: チャンネルを作成する」と「ステップ 8: 望ましくないコンテンツや視聴者を防ぐ」を更新しました。
- [Service Quotas](#) - 新しいエンドポイントの TPS 制限を追加し、「その他のクォータ」に新しいクォータを追加しました。
- [望ましくないコンテンツと視聴者](#) - 「再生制限ポリシーの使用」を追加しました。
- [プライベートチャンネル](#) - コンソールナビゲーションペインの再生キーの場所を更新しました。

[「API の変更」](#)も参照してください。

オーディオのみの再生	プレイヤーの概要に オーディオのみの再生 を追加しました。	2024 年 1 月 25 日
Player SDK 1.24.0	Player SDK ガイド (Web 、 Android 、 iOS 、 Video.js 統合 、および JW Player 統合) で、新しいリリースのバージョン番号とアーティファクトリンクを更新しました。 。 Amazon IVS ドキュメントのランディングページ では、Player SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。このリリースについては、「 Amazon IVS リリースノート 」も参照してください。 ウェブガイドで、新しいセクション「オーディオのみの再生」を追加し、audio-only レンディションのサポート不足に関する「既知の問題」を削除しました。	2024 年 1 月 18 日
Amazon S3 への自動録画のトラブルシューティング	「トラブルシューティング」に、「 SKMS-S3S3 暗号化を使用できますか? 」というセクションを追加しました。	2024 年 1 月 4 日

[Broadcast SDK: Android 1.13.4、iOS 1.13.4、Web 1.7.0](#)

2024 年 1 月 3 日

Broadcast low-latency-streaming SDK ガイド: <https://docs.aws.amazon.com/ivs/latest/lowLatencyUserGuide/broadcast-ios.html>、iOS、および [Web](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。[Amazon IVS ドキュメントのランディングページ](#)では、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。このリリースについては、「[Amazon IVS リリースノート](#)」も参照してください。

[チャット UG を分割する](#)

2023 年 12 月 28 日

このリリースに伴い、ドキュメントに大幅な変更が行われました。チャット情報を IVS 低レイテンシーストリーミングユーザーガイドから、[IVS ドキュメントのランディングページ](#)の既存の IVS Chat セクションにある新しい IVS Chat ユーザーガイドに移動しました。

その他のドキュメントの変更については、「[ドキュメント履歴 \(チャット\)](#)」を参照してください。

[IVS の用語集](#)

用語集を拡張し、IVS のリアルタイム、低レイテンシー、チャット用語を網羅しました。

2023 年 12 月 20 日

[IAM マネージドポリシー](#)

IVS ReadOnlyAccess と IVS の 2 つのマネージドポリシーを追加しました FullAccess。以下を参照してください。

2023 年 12 月 5 日

- セキュリティページの [Amazon IVS 用マネージドポリシー](#) に関する新しいセクション。
- 「IVS 低レイテンシーストリーミングを開始する」の「[ステップ 3: IAM アクセス権限の設定](#)」への変更。

[Broadcast SDK: Android 1.13.2、iOS 1.13.2](#)

Broadcast low-latency-streaming SDK ガイド: [Android](#) および [iOS](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2023 年 12 月 4 日

[Amazon IVS ドキュメントのランディングページ](#)

は、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「[Amazon IVS リリースノート](#)」も参照してください。

[Broadcast SDK: Android](#)

[1.13.1](#)

Broadcast low-latency-streaming SDK ガイド: [Android](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2023 年 11 月 21 日

[Amazon IVS ドキュメント](#) [のランディングページ](#)

は、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[Broadcast SDK: Android](#)

[1.13.0、iOS 1.13.0](#)

Broadcast low-latency-streaming SDK ガイド: [Android](#) および [iOS](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2023 年 11 月 17 日

[Amazon IVS ドキュメント](#) [のランディングページ](#)

は、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[S3 への自動録画](#)

「[フラグメント化されたストリームを結合](#)」の「利用資格」に、「各ストリームは前のストリームから 10 秒以上後に開始する必要があります」という箇条書きを追加しました。

2023 年 11 月 17 日

[サーバーサイドコンポジションとリアルタイムコンポジションの録画](#)

「[Amazon IVS ストリームで複数ホストを有効にする](#)」に、「ステージのブロードキャスト: クライアントサイドとサーバーサイドのコンポジション」を追加し、「4. ステージをブロードキャストする」を更新しました。

2023 年 11 月 16 日

「[セキュリティ](#)」で、「ID ベースのポリシーの例」の「Amazon IVS コンソールを使用する」のポリシーに、S3 エンドポイントを追加しました。

その他の変更については、「[ドキュメント履歴 \(リアルタイムストリーミング\)](#)」を参照してください。

[Player SDK 1.23.0](#)

Player SDK ガイド
([Web](#)、[Android](#)、[iOS](#)、[Video.js](#)
[統合](#)、および [JW Player 統
合](#)) で、新しいリリースのバー
ジョン番号とアーティファク
トリンクを更新しました。

2023 年 11 月 14 日

[Amazon IVS ドキュメント
のランディングページ](#)
で
は、Player SDK リファレンス
のリンクが更新され、新しい
バージョンを表示するよう
になりました。

このリリースについては、
「[Amazon IVS リリースノー
ト](#)」も参照してください。

[IVS Player と Broadcast SDK](#)

「[Amazon IVS Player SDK](#)」
と「[IVS Broadcast SDK \(低レ
イテンシーストリーミング\)](#)」
で、サポートされている SDK
のバージョンが明確になるよ
うに、「プラットフォームの
要件」を「ネイティブプラッ
トフォーム」に更新しまし
た。

2023 年 11 月 9 日

[IVS 低レイテンシーストリー ミングを開始する](#)

「[IAM アクセス許可を設定
する](#)」の手順が更新されまし
た。

2023 年 10 月 20 日

[Broadcast SDK: Web 1.6.0](#)

Broadcast low-latency-streaming SDK ガイド: [Web](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2023 年 10 月 16 日

[Amazon IVS ドキュメントのランディングページ](#)は、ブロードキャスト SDK リファレンスの最新バージョンを参照しています。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

ウェブガイドの MediaStream 「デバイスから を取得する」で、2 つのmax行も削除されました。ベストプラクティスは のみを指定することで ideal。

[IVS 低レイテンシーストリーミングのモニタリング](#)

「ライブストリームヘルスのモニタリング」ページの名前を変更し、「による IVS のモニタリング」の情報を追加しました CloudWatch (別のページとして削除されました)。CloudWatch コンソールの手順を更新しました。

2023 年 10 月 12 日

[Broadcast SDK: Android](#)

[1.12.1](#)

Broadcast low-latency-streaming SDK ガイド: [Android](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。また、「[Bluetooth マイクの使用](#)」という新しいセクションも追加されました。

2023 年 10 月 12 日

[Amazon IVS ドキュメント](#) [のランディングページ](#)

は、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「[Amazon IVS リリースノート](#)」も参照してください。

[Player SDK 1.22.0](#)

Player SDK ガイド ([Web](#)、[Android](#)、[iOS](#)、[Video.js](#) [統合](#)、および [JW Player 統合](#)) で、新しいリリースのバージョン番号とアーティファクトリンクを更新しました。

2023 年 10 月 3 日

[Amazon IVS ドキュメント](#) [のランディングページ](#)

は、Player SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「[Amazon IVS リリースノート](#)」も参照してください。

コンソール内ストリーミング	「Getting Started with Low-Latency Streaming」の「 ステップ 5: ストリーミングソフトウェアの設定 」にコンソール内ストリーミングを追加しました。	2023 年 10 月 2 日
Broadcast SDK: ミキサーガイド	「 Broadcast のミラーリング 」を追加しました (Android と iOS の例を使用)。	2023 年 9 月 18 日
Broadcast SDK: Web 1.5.2	Broadcast low-latency-streaming SDK ガイド: Web で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。 Amazon IVS ドキュメントのランディングページ は、ブロードキャスト SDK リファレンスの最新バージョンを参照しています。 このリリースについては、「Amazon IVS リリースノート 」も参照してください。	2023 年 9 月 14 日
望ましくないコンテンツ	「トラブルシューティング FAQ」の既存のコンテンツを独自のトップレベルページに分割します。 IVS 低レイテンシーストリーミングの開始 に、「 ステップ 8: 望ましくないコンテンツを防ぐ (推奨) 」を追加しました。	2023 年 9 月 8 日

[Amazon S3 への自動録画](#)

[バイト範囲プレイリスト](#)に、セグメントの再生時間がストリームに設定されたキーフレーム間隔と同じであることを明記しました (約 2 秒の固定時間ではありません)。

2023 年 8 月 25 日

[Broadcast SDK: Web 1.5.1、Android 1.12.0、iOS 1.12.0](#)

Broadcast low-latency-streaming SDK ガイド: [Web https://docs.aws.amazon.com/ivs/latest/LowLatencyUserGuide/broadcast-android.html](https://docs.aws.amazon.com/ivs/latest/LowLatencyUserGuide/broadcast-android.html)、Android、[iOS](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2023 年 8 月 23 日

[Amazon IVS ドキュメントのランディングページ](#)

は、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

Player SDK 1.21.0

Player SDK ガイド
([Web](#)、[Android](#)、[iOS](#)、[Video.js](#)
[統合](#)、および [JW Player 統
合](#)) で、新しいリリースのバー
ジョン番号とアーティファク
トリンクを更新しました。

2023 年 8 月 22 日

[Amazon IVS ドキュメント
のランディングページ](#)
で
は、Player SDK リファレンス
のリンクが更新され、新しい
バージョンを表示するよう
になりました。

このリリースについては、
「Amazon IVS [リリースノー
ト](#)」も参照してください。

チャンネルタイプの定義

チャンネルタイプの定義を更新
して、特にレンダリングされ
たトランスコードラダーに関
する詳細が追記されました。
「IVS ストリーミング設定」
の[チャンネルタイプ](#)を参照して
ください。

2023 年 8 月 18 日

[リアルタイムストリーミングのローンチ](#)

2023 年 8 月 7 日

このリリースに伴い、ドキュメントに大幅な変更が行われました。以前のドキュメントの名前を「IVS 低レイテンシーストリーミング」に変更し、新しい IVS リアルタイムストリーミングドキュメントを公開しました。「[IVS ドキュメントのランディングページ](#)」に、リアルタイムストリーミングと低レイテンシーストリーミングの個別のセクションが追加されました。各セクションには、それぞれのユーザーガイドと API リファレンスがあります。

一部の情報を「IVS 低レイテンシーユーザーガイド」から新しい「IVS リアルタイムユーザーガイド」に移動しました。

- ステージと複数のホストに関する詳細の大部分。
- 「ステージの健康度のモニタリング」は「[リアルタイムストリーミングのモニタリング](#)」になりました。

その他のドキュメントの変更については、以下を参照してください。

- [ステージ API リファレンスの変更](#)

- [ドキュメント履歴 \(リアルタイムストリーミング\)](#)

[Broadcast SDK: Web 1.5.0、Android 1.11.0、iOS 1.11.0](#)

ブロードキャスト SDK ガイド: [Web](#)、[Android](#)、および [iOS](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2023 年 8 月 7 日

[Amazon IVS ドキュメントのランディングページ](#)

は、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[プライベートチャネルの設定](#)

[\[トークンスキーマ\]](#) で、exp フィールドに関する明確な情報を追加しました。

2023 年 7 月 31 日

[セキュリティ: IVS ステータス情報の取得](#)

「[インシデント対応](#)」で、IVS ステータスの取得に関する情報を更新し、AWS Health Dashboard を参照するようにしました。

2023 年 7 月 31 日

[Amazon S3 への自動録画: OAC と CORS](#)

「[プライベートバケットからの録画コンテンツの再生](#)」で、オリジンアクセスアイデンティティ (OAI) をオリジンアクセスコントロール (OAC) に置き換えました。また、録画されたストリームを再生するための CORS 用の S3 バケットの設定に関する情報も追加しました。

2023 年 7 月 31 日

[リソースとサポート](#)

「[パートナーソリューション](#)」>「[フェイスフィルターと背景フィルター](#)」に、Camera Kit に関する段落を追加しました。

2023 年 7 月 21 日

[Broadcast SDK: Android ガイド](#)

小規模な変更。導入部で、エミュレーターはサポートされていないことに言及しました。「[プレイヤーの作成とイベントリスナーのセットアップ](#)」で、PlayerActivity class を Activity に変更しました。「[スレッドの安全性](#)」でテキストを変更しました。

2023 年 7 月 21 日

[R2S3 レンディションファイルタリングおよびサムネイルの強化](#)

2023 年 7 月 17 日

IVS で、Amazon S3 への録画時にストリームに対して生成されるレンディションと、サムネイル用に生成される解像度を制御できるようになりました。「IVS ユーザーガイド」で以下を参照してください。

- 「[IVS の開始](#)」 — 「ステップ 4: チャンネルを作成する」 > 「コンソールの説明」で、スクリーンショットと説明を更新しました。
- 「[Amazon S3 への自動録画](#)」 — 「JSON メタデータファイル」に latest_thumbnail を追加し、thumbnail を更新しました。「サムネイル」と「録画のレンディションを探す」に、レンディション解像度の説明を追加しました。
- 「[コスト](#)」 — 「録画したビデオの保存」で、スクリーンショットを更新しました。

「[IVS API リファレンスの変更](#)」も参照してください。

[Player SDK 1.20.0](#)

すべてのプレーヤーガイド ([Web](#)、[Android](#)、[iOS](#)、[Video.js](#) [統合](#)、および [JW Player](#) [統合](#)) で、新しいリリースのバージョン番号とアーティファクトリンクを更新しました。

2023 年 7 月 14 日

[Amazon IVS ドキュメント](#) の [ランディングページ](#) では、Player SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「[Amazon IVS リリースノート](#)」も参照してください。

[IVS の開始方法](#)

「[録画を無効にする方法](#)」で、CLI の例を修正しました。

2023 年 7 月 14 日

[Broadcast SDK: Web](#)
[1.4.0、Android 1.10.0、および](#)
[iOS 1.10.0](#)

ブロードキャスト SDK ガイド: [Web](#)、[Android](#)、および [iOS](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2023 年 7 月 13 日

[Amazon IVS ドキュメント](#)
[のランディングページ](#)では、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「[Amazon IVS リリースノート](#)」も参照してください。

[プライベートチャンネルの ビューアーセッションの取り 消し](#)

IVS のユーザーは、認証トークンに関連付けられたビューアーセッションを取り消して、そのトークンを使用した再生を禁止または停止できるようになりました。詳細については、「[プライベートチャンネルの設定](#)」を参照してください。

2023 年 6 月 28 日

- 「トークンスキーマ」
— viewer-id を追加し、viewer-session-version を変更しました。
- 「ビューアーセッションの取り消し」— 新しいセッション。

「[IVS API リファレンスの変更](#)」も参照してください。

[セキュリティの TLS 更新](#)

[インフラストラクチャセキュリティ] > [API コール](#) で、TLS バージョンを最小 1.2、推奨 1.3 に更新しました。

2023 年 6 月 27 日

[Broadcast SDK: iOS 1.9.1 および iOS 1.7.5](#)

Broadcast SDK ガイド: [iOS](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2023 年 6 月 27 日

[Amazon IVS ドキュメントのランディングページ](#)

[このページ](#)は、Broadcast SDK リファレンスの最新バージョンを参照しています。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[Broadcast SDK: Web 1.3.3](#)

Broadcast SDK ガイド: [Web](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2023 年 6 月 16 日

[Amazon IVS ドキュメントのランディングページ](#)

[このページ](#)は、Broadcast SDK リファレンスの最新バージョンを参照しています。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[Advanced チャンネルタイプ](#)

2023 年 6 月 2 日

ADVANCED_SD と ADVANCED_HD の 2 つの新しいチャンネルタイプが導入されました。次のページを更新しました。

- [Player SDK の概要](#) – 「サードパーティ製プレーヤーのレイテンシーの削減」に、Advanced ストリームではレイテンシー削減機能が必要ないことを記載しました。
- [Broadcast Web SDK ガイド](#) – 「AmazonIVSBroadcastClient のインスタンスを作成する」を変更しました。
- [Broadcast Android SDK ガイド](#) – 「推奨されるブロードキャスト設定の取得」を変更しました。
- [Broadcast iOS SDK ガイド](#) – 「推奨されるブロードキャスト設定の取得」を変更しました。
- [Service Quotas](#) – 「その他のクォータ」にある「IVS」に、新しいチャンネルタイプの「取り込みビットレート」に関する 2 行を追加しました。
- [ストリーミング設定](#) – 「チャンネルタイプ」を変更しました。
- [コスト](#) – 新しいチャンネルタイプと、「選択のヘルプ」

ツールの説明を追加しました

[Broadcast SDK: iOS 1.9.0 および iOS 1.9.0](#)

[Android](#) および [iOS](#) 向けの Broadcast SDK ガイドで、新しいリリースのバージョン番号とアーティファクトリンクを更新しました。

2023 年 6 月 1 日

[Amazon IVS ドキュメントのランディングページ](#)

は、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

[「Broadcast SDK の概要」](#)

で、サポート対象の iOS バージョンを 11 以上から 12 以上に更新しました (ステージ機能がない SDK 用)。

「[iOS ガイド](#)」に、「iOS がカメラの解像度とフレームレートを選択する方法」という新しいセクションを追加しました。

このリリースについては、「[Amazon IVS リリースノート](#)」も参照してください。

[Amazon S3 への自動録画](#)

「例: recording_ended.json」で、byte_range_playlist 値を byte-range-multivariant.m3u8 から byte-range-variant.m3u8 更新しました。

2023 年 5 月 25 日

[Player SDK 1.19.0](#)

すべてのプレーヤーガイド ([Web](#)、[Android](#)、[iOS](#)、[Video.js](#) [統合](#)、および [JW Player 統合](#)) で、新しいリリースのバージョン番号とアーティファクトリンクを更新しました。

2023 年 5 月 23 日

[Amazon IVS ドキュメントのランディングページ](#)

は、Player SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「[Amazon IVS リリースノート](#)」も参照してください。

[Broadcast SDK: iOS 1.8.1 および iOS 1.7.4](#)

Broadcast SDK ガイド: [iOS](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2023 年 5 月 16 日

[Amazon IVS ドキュメントのランディングページ](#)

[ジ](#)は、Broadcast SDK リファレンスの最新バージョンを参照しています。

このリリースについては、「[Amazon IVS リリースノート](#)」も参照してください。

ステージ健康度のモニタリング

2023 年 5 月 11 日

新しい Amazon IVS 機能の新しいユーザーガイドページとして、「[ステージ健康度のモニタリング](#)」を追加しました。ステージのヘルスについては、次の更新もあります。

- 「[Amazon IVS ストリームで複数ホストを有効にする](#)」にヘルス情報が追加されました。
- Amazon [IVS](#) での Amazon の使用に 2 つのステージ更新イベントを追加 [EventBridge](#) しました。
- 「[Amazon IVS Service Quotas](#)」に新しいエンドポイントのコールレートクォータが追加されました。

注: 2023 年 8 月 2 日の IVS リアルタイムストリーミングのローンチに伴い、このドキュメントの名前は「Amazon IVS Real-Time Streaming のモニタリング」に変更され、新しい「IVS リアルタイムストリーミングユーザーガイド」に移動しました。

[ステージ参加者の制限](#)

「[Amazon IVS Service Quotas](#)」で、「ステージ参加者」の制限が削除されました。これに代わり、サブスクライバーとパブリッシャーの参加者が制限されます。

2023 年 5 月 2 日

[Broadcast SDK: Web 1.3.2](#)

Broadcast SDK ガイド: [Web](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2023 年 5 月 1 日

[Amazon IVS ドキュメントのランディングページ](#)

[このページ](#)は、Broadcast SDK リファレンスの最新バージョンを参照しています。

このリリースについては、「[Amazon IVS リリースノート](#)」も参照してください。

[RTMP サポート:ドキュメントの正誤表](#)

「[Broadcast Android SDK ガイド](#)」と「[Broadcast iOS SDK ガイド](#)」を変更して、これらの SDK が RTMPS 取り込みのみをサポートしている (安全ではない RTMP 取り込みはサポートしていない) ことを示しました。

2023 年 4 月 29 日

[ステージ参加者の制限](#)

このリリースでは、以下が変更されています。

2023 年 4 月 27 日

- [複数のホストを有効にする](#) — ステージ参加者の最大数を 12 人から 1,000 人に更新しました。
- [Service Quotas](#) — 参加者の上限を 1,000 人に更新し、サブスクライバーとパブリッシャーの参加者に対する新しい制限を追加しました。一部のエンドポイントの TPS を変更します。

[IVS ユーザーガイドのランディングページ](#)

「[Amazon Interactive Video Service とは何ですか?](#)」ホームページに「複数ホスト」と「IVS Chat」のセクションを追加し、「レイテンシー」のセクションを更新しました。

2023 年 4 月 27 日

[リソースとサポート](#)

「パートナーソリューション」>「フェイスフィルターと背景フィルター」で、DeepAR リンクが更新されました。

2023 年 4 月 25 日

[リソースとサポート](#)

パートナーソリューションにセクションを追加しました。

2023 年 4 月 17 日

[Player SDK: Web ガイド](#)

「既知の問題と回避策」に問題 (Web Player が audio_only レンディションをサポートしていない) が追加されました。

2023 年 4 月 17 日

[ストリーミング設定](#)

[クローズドキャプション](#)に、キャプションに関する新しいブログ記事へのリンクを追加しました。

2023 年 4 月 14 日

[Broadcast Web SDK ガイド](#)

その他の更新は、以下の通りです。

2023 年 4 月 10 日

- 「AmazonIVSBroadcastClient のインスタンスを作成する」に、クライアント側の設定がバックエンドのチャンネルタイプと一致していることを確認するためのメモを追加しました。
- 「動画を非表示にする」のコード例では、`VIDEO_DEVICE_NAME` は `VIDEO_DEVICE_NAME.source` へ変更されました。
- 「複数ホストを有効にする」で、`ConnectionState` 参照を `StageConnectionState` に変更しました。
- 「Broadcast SDK を使用して複数のホストを追加する」と「既知の問題」で、この情報を同期しました [GitHub](#)。

[ストリーミング設定](#)

[「ビデオ設定」](#)に、`ColorSpace` 箇条書きを追加しました。

2023 年 4 月 5 日

[複数ホストを有効にする](#)

[AWS CLI のセットアップ](#)

2023 年 4 月 5 日

プで、[ivsrealtime](#) から [ivs-realtime](#) へステージ名前空間を変更しました。

[Player SDK 1.18.0](#)

すべてのプレーヤーガイド ([Web](#)、[Android](#)、[iOS](#)、[Video.js](#) [統合](#)、および [JW Player 統合](#)) で、新しいリリースのバージョン番号とアーティファクトのリンクを更新しました。

2023 年 4 月 4 日

[Amazon IVS ドキュメントのランディングページ](#)

は、Player SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[RTMP サポート](#)

一部のドキュメントで、RTMP (安全でない取り込み) と RTMPS の両方がサポートされるようになったことを明示しました。特に、これは取り込みエンドポイントに影響します。「[ストリーミング ソフトウェアのセットアップ](#)」、「[Broadcast Android SDK ガイド](#)」、および「[Broadcast iOS SDK ガイド](#)」を参照してください。

2023 年 3 月 30 日

プライベートチャンネルの設定	<p>「再生トークンの生成と署名」で、一時使用トークンを生成するためのオプションフィールド <code>single-use-uuid</code> がペイロードに追加されました。</p>	2023 年 3 月 29 日
Broadcast SDK: Web 1.3.1	<p>Broadcast SDK ガイド: Web で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。</p> <p>Amazon IVS ドキュメントのランディングページでは、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。</p> <p>このリリースについては、「Amazon IVS リリースノート」も参照してください。</p>	2023 年 3 月 28 日
ストリーム上の複数ホストをサポート	<p>新規ページ「IVS ストリームでの複数ホストの有効化」を追加しました。また、Service Quotasに「Amazon IVS ステージ」エンドポイントを追加し、[その他のクォータ] > [Amazon IVS] にステージ制限を追加しました。</p> <p>「ステージ API リファレンスの変更」も参照してください。</p>	2023 年 3 月 23 日

[Broadcast SDK: Android 1.8.0、iOS 1.8.0、および Web 1.3.0](#)

ブロードキャスト SDK ガイド: [Android](#)、[iOS](#)、および [Web](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2023 年 3 月 23 日

[Amazon IVS ドキュメントのランディングページ](#)で

は、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

[Broadcast SDK の概要](#)に、ステージプラットフォームの要件を追加しました。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[Web Broadcast SDK](#)

[既知の問題と回避策](#)に、Safari ブロードキャストの視聴者のビデオフィードに緑色のアーティファクトが表示されることがあるという問題を追加しました。

2023 年 3 月 17 日

[Broadcast SDK: Android 1.7.3](#)

ブロードキャスト SDK ガイド: [Android](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2023 年 3 月 2 日

リポジトリの [Amazon IVS ドキュメントのランディングページ](#) では、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを参照するようになりました。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[Player SDK 1.17.0](#)

すべてのプレーヤーガイド ([Web](#)、[Android](#)、[iOS](#)、[Video.js](#) [統合](#)、および [JW Player](#) [統合](#)) で、新しいリリースのバージョン番号とアーティファクトリンクを更新しました。

2023 年 2 月 28 日

[Amazon IVS ドキュメントのランディングページ](#) では、Player SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[Service Quotas](#)

すべてのクォータがリージョンごとに適用されることを明確にしました。

2023 年 2 月 24 日

[トラブルシューティングに関するよくある質問](#)

「プライベートチャンネルを使用する」では、望ましくないコンテンツを防ぐためにプライベートチャンネルの使用について明確化しました。
「ブロードキャストとエンコーディング」に、IVS Web Broadcast SDK セッションのトラブルシューティングと Chrome WebRTC-internals の使用に関するサブセクションを 2 つ追加しました。

2023 年 2 月 17 日

[S3 への自動録画用のバイト範囲タグとマニフェストファイル](#)

「[Amazon S3 への自動録画](#)」で、「録画の内容」が更新され、「バイト範囲プレイリスト」が追加され、recording_started と recording_ended の JSON の例に新しいフィールドが追加されました。

2023 年 2 月 16 日

[IVS Chat の開始方法](#)

冒頭で、IVS Chat はビデオストリームなしで単独でも使用できることを記載しました。
「Amazon IVS Chat ユーザーガイド」の「[IVS Chat の開始方法](#)」を参照してください。

2023 年 2 月 9 日

[トラブルシューティングに関するよくある質問](#)

「望ましくないコンテンツ」に関する新しいセクションを追加しました。

2023 年 2 月 6 日

2023 年 9 月 8 日更新: このセクションは [望ましくないコンテンツ](#) に移動されました。

Player SDK の概要	「 ブラウザとプラットフォームの要件 」に、Web SDK Video.js と Player JW の統合はブラウザのような環境ではサポートされていない旨の注記を追加しました。	2023 年 2 月 6 日
Amazon S3 への自動録画	フラグメント化されたストリームのマージに関する「 対象 」の要件で、必要なビットレートの差を 10% から 50% に変更しました。	2023 年 2 月 6 日
ストリーミング設定	「 Amazon IVS Broadcast SDK を使用してストリーミングする 」を改定し、(Android と iOS だけでなく) Web Broadcast SDK を含めました。	2023 年 2 月 2 日

[IVS Chat Client Messaging SDK: Android 1.1.0](#)

Chat SDK ガイド: [Android](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2023 年 1 月 31 日

[Amazon IVS ドキュメントのランディングページ](#)は、SDK リファレンスの最新バージョンを参照しています。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

このリリースには、Chat Kotlin コルーチンの広範なチュートリアルが含まれており、次の 2 つの部分に分かれています。

- [パート 1: チャットルーム](#)
- [パート 2: メッセージとイベント](#)

[Chat Android SDK のチュートリアル](#)

Chat Client Messaging SDK 用に広範な Android のチュートリアルを追加しました。このチュートリアルは次の 2 つの部分に分かれています。

2023 年 1 月 24 日

- [パート 1: チャットルーム](#)
- [パート 2: メッセージとイベント](#)

Service Quotas

一部の Chat のクォータを引き上げました。 2023 年 1 月 19 日

- CreateChatToken、DeleteMessage DisconnectUser、および SendEvent ルームの TPS
- その他のクォータ: 同時チャット接続、DeleteMessage、SendMessage リクエストのレート DisconnectUser、接続あたりのメッセージングリクエストのレート、ルーム

プライベートチャンネル

「トークンスキーマ」で、strict-origin-enforcement フィールドをトークンペイロードに追加しました。 2023 年 1 月 17 日

[Player SDK 1.16.0](#)

すべてのプレーヤーガイド ([Web](#)、[Android](#)、[iOS](#)、[Video.js](#) [統合](#)、および [JW Player 統合](#)) で、新しいリリースのバージョン番号とアーティファクトリンクを更新しました。

2023 年 1 月 17 日

[Amazon IVS ドキュメント](#) [のランディングページ](#)で

は、Player SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「[Amazon IVS リリースノート](#)」も参照してください。

[Chat React と React Native のベストプラクティス](#)

この新しい Chat ページを追加しました。

2023 年 1 月 13 日

注: 2023 年 12 月 28 日に、[このドキュメント](#)は新しい IVS Chat ユーザーガイドに移動されました。

[Chat React Native SDK チュートリアル](#)

チャットクライアントメッセージング SDK 用に広範な React Native のチュートリアルを追加しました。このチュートリアルは次の 2 つのパートに分かれています。

2023 年 1 月 10 日

- [パート 1: チャットルーム](#)
- [パート 2: メッセージとイベント](#)

トラブルシューティング	ベストプラクティスとトラブルシューティングのヒントを記載した、新しいトラブルシューティングに関するよくある質問のページを追加しました。	2023 年 1 月 6 日
record-to-S3 のマニフェストファイルにタイムスタンプを追加しました	S3 への自動録画機能で作成した S3 マニフェストファイルに、タイムスタンプを追加しました。Amazon IVS リリースノート を参照してください。	2022 年 12 月 9 日
Player SDK のレイテンシー	サードパーティ製プレイヤーのレイテンシーを低減 しました。	2022 年 12 月 8 日
Broadcast Web SDK ガイド	このページにコンテンツを追加しました (以前はのみでした GitHub)。	2022 年 12 月 8 日
Broadcast SDK: Android 1.7.2	<p>ブロードキャスト SDK ガイド: Android で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。</p> <p>リポジトリの Amazon IVS ドキュメントのランディングページでは、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを参照するようになりました。</p> <p>このリリースについては、「Amazon IVS リリースノート」も参照してください。</p>	2022 年 12 月 6 日

[IVS の設定](#)

[Amazon Interactive Video](#)

2022 年 12 月 5 日

[Service の使用開始](#)で、AWS アカウントを作成してアクセス許可を設定する手順を更新しました。「Step 2: Set Up Root and Administrative Users」(ステップ 2: ルートユーザーと管理ユーザーの設定)を追加しました。

[セキュリティ](#)で、IAM セクションの冒頭に小さな変更を加えました。

[チャット: セットアップと iOS SDK のチュートリアル](#)

[IVS チャットの開始方法](#)で

2022 年 12 月 5 日

「初期設定」を更新し、名前を変更しました。

[「ユーザーガイド」に「チャット iOS チュートリアル」](#)ページを追加し、の既存のチュートリアルを参照しました GitHub。

[S3 への自動録画のコスト](#)

[Amazon S3 への自動録画](#)でコストを明確にしました。

2022 年 12 月 2 日

[Chat JavaScript SDK チュートリアル](#)

チャットクライアントメッセージング SDK 用に広範な JS のチュートリアルを追加しました。このチュートリアルは次の 2 つのパートに分かれています。

2022 年 12 月 2 日

- [パート 1: チャットルーム](#)
- [パート 2: メッセージとイベント](#)

Web Player の既知の問題

Player ウェブ SDK ガイドに、[既知の問題と回避方法](#)を追加しました。iOS モバイルブラウザでミュート状態のライブストリームを再生中に、非アクティブなプレイヤータブを再開するとプレイヤーが不安定になることがあります。

2022 年 11 月 18 日

プライベートチャンネルの設定

「再生キーの作成またはインポート」で、内容を再編成し、プライベートキーとパブリックキーの使用方法を明確にしました。「再生トークンの生成と署名」で、jwt.io でパブリックキーの入力が必須ではないことを明確にしました。

2022 年 11 月 18 日

[チャットのログ記録](#)

2022 年 11 月 17 日

この新機能の初回リリース。
次のユーザーガイドの変更点
を参照してください。

- [チャットのログ記録](#) — 新しいページ。
- [チャットの開始方法](#) — IAM のアクセス許可を更新し、チャットのログ記録を設定する手順を追加しました。
- [Service Quotas](#) — 新しいエンドポイントとログ記録設定に関する制限を追加しました。
- Cloudwatch — ログ送信先のメトリクスを追加しました。

2023 年 10 月 12 日の更新:
この CloudWatch ドキュメントは削除され、コンテンツは「[IVS 低レイテンシーストリーミングのモニタリング](#)」に移動されました。

2023 年 12 月 28 日の更新:
チャット関連の CloudWatch コンテンツは「[Amazon IVS Chat のモニタリング](#)」に移動されました。

[Chat Client Messaging SDK: JavaScript 1.0.2](#)

Chat SDK ガイド: で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました[JavaScript](#)。

2022 年 11 月 9 日

[Amazon IVS ドキュメントのランディングページ](#)は、SDK リファレンスの最新バージョンを参照しています。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[ライブチャネルの分割ビュー \(ライブストリームのヘルスのモニタリング用\)](#)

[ストリームセッションデータへのアクセス](#)に、新しい分割ビューにアクセスするためのコンソールでの手順を追加しました。これは「ライブチャネル」ページからセッション状態のデータを直接、そしてすばやく取得する方法です。

2022 年 11 月 8 日

[リソースとサポート](#)

開発者コミュニティサイトに IVS ブログへのリンクを追加しました。

2022 年 11 月 7 日

[Amazon S3 への自動録画](#)

「断片化されたストリームの結合」の「[対象](#)」で、「Source video quality must be the same」(ソースビデオの品質が同じであること) という冗長な箇条書きを削除しました。

2022 年 11 月 7 日

[Player SDK 1.14.0](#)

すべてのプレーヤーガイド ([Web](#)、[Android](#)、[iOS](#)、[Video.js](#) [統合](#)、および [JW Player](#) [統合](#)) で、新しいリリースのバージョン番号とアーティファクトリンクを更新しました。

2022 年 11 月 1 日

[Amazon IVS ドキュメント](#) [のランディングページ](#)で

は、Player SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「[Amazon IVS リリースノート](#)」も参照してください。

[Player SDK: Web ガイド](#)

新しい CSP ルールに対応するように、すべてのブラウザの新しいバージョンが更新されたことを反映して、[コンテンツセキュリティポリシーの操作](#)を更新しました。「同一オリジンでのアセットのホスティング」と「別のオリジンでのアセットのホスティング」に関する古いセクションを削除しました。

2022 年 10 月 27 日

[Amazon IVS Chat の開始方法](#)

以前は「チャットクライアントの認証と承認」だったステップ 3 を更新し、明確化しました。[チャットトークンを作成する](#)ようになりました。

2022 年 10 月 27 日

[Player SDK: Web ガイド](#)

「サンプルコード」では、PLAYBACK_URL 前後に引用符を追加し、URL 文字列に置き換える必要があることを明確にしました。

2022 年 10 月 24 日

[チャットクライアントメッセージング SDK: JavaScript TAK](#)

[React Native Support](#) という新しいセクションを追加しました。

2022 年 10 月 24 日

[IVS Chat Client Messaging SDK: JavaScript 1.0.1](#)

この新しい SDK の初回リリース。「IVS ユーザーガイド」の「[Amazon IVS Chat Client Messaging SDK](#)」を参照してください。

2022 年 10 月 18 日

[Amazon IVS ドキュメントのランディングページ](#)は、SDK リファレンスの最新バージョンを参照しています。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[Broadcast SDK: iOS 1.7.1](#)

Broadcast SDK ガイド: [iOS](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2022 年 10 月 6 日

[Amazon IVS ドキュメントのランディングページ](#)

は、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[Web Player SDK 1.13.0 リリースノート](#)

ウェブプレーヤー 1.13.0 のリリースノートに、Sawmill Enabled ログに関する既知の問題が追加されました。

2022 年 9 月 27 日

[Broadcast SDK 1.7.0 リリース](#)

ブロードキャスト SDK ガイド: [Android](#)、[iOS](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2022 年 9 月 22 日

[Amazon IVS ドキュメントのランディングページ](#)

は、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[プレイヤー 1.13.0 リリース](#)

すべてのプレイヤーガイド ([Web](#)、[Android](#)、[iOS](#)、[Video.js](#) [統合](#)、および [JW Player](#) [統合](#)) で、新しいリリースのバージョン番号とアーティファクトリンクを更新しました。

2022 年 9 月 20 日

[Amazon IVS ドキュメントのランディングページ](#)では、Player SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「[Amazon IVS リリースノート](#)」も参照してください。

[Broadcast SDK: iOS 1.5.2](#)

Broadcast SDK ガイド: [iOS](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2022 年 9 月 12 日

[Amazon IVS ドキュメントのランディングページ](#)は、ブロードキャスト SDK リファレンスの最新バージョンを参照しています。

このリリースについては、「[Amazon IVS リリースノート](#)」も参照してください。

[IVS Chat Client Messaging SDK: Android 1.0.0 および iOS 1.0.0](#)

新しい SDK の初回リリース。「IVS ユーザーガイド」の「[Amazon IVS Chat Client Messaging SDK](#)」を参照してください。

2022 年 9 月 8 日

[Amazon IVS ドキュメントのランディングページ](#)は、SDK リファレンスの最新バージョンを参照しています。

このリリースについては、「[Amazon IVS リリースノート](#)」も参照してください。

「[Amazon IVS Chat の開始方法](#)」を更新し、さまざまなデモ (トークン生成のデモが実行できるバックエンドサーバーアプリを含む) へのリンクと、チャットメッセージを削除するためのサンプルコードを追加しました。

[Amazon による Amazon IVS のモニタリング CloudWatch](#)

Channel デイメンションのある Amazon IVS のメトリクスの一部の説明が修正されました。チャンネル値は ARN ではありません (前述の通り)。これはチャンネルの resource-id であり、ARN の最後の部分です。

2022 年 9 月 2 日

2023 年 10 月 12 日の更新:
この CloudWatch ドキュメントは削除され、コンテンツは「[IVS 低レイテンシーストリーミングのモニタリング](#)」に移動されました。

[リソースとサポート](#)

「Amazon IVS ユーザーガイド」に新しいページを追加しました。これは Amazon IVS に関する追加情報およびサポートを示しています。

2022 年 9 月 1 日

断片化されたストリームの結合

この新機能の初回リリース。これらのドキュメントの変更を参照してください。

2022 年 8 月 30 日

- Amazon IVS の開始方法 - 「[ステップ 3: 任意の録画によるチャンネルの作成](#)」のコンソールと CLI の手順を更新しました。
- S3 への自動録画 - 「[断片化されたストリームの結合](#)」が追加されました。
- EventBridge - 「例: 記録状態の変更」に recording_session_id および recording_session_stream_ids フィールドを追加しました。
<https://docs.aws.amazon.com/ivs/latest/userguide/eventbridge.html#eventbridge-examples-recording-state-change>

ライブストリームのヘルスのモニタリング

「[ヘルスによるストリームのフィルタリング](#)」で、CLI の例を修正し、filter-by name を filter-by health へ変更しました。

2022 年 8 月 17 日

BASIC チャンルの拡張

2022 年 8 月 16 日

BASIC チャンルの最大解像度とビットレートが変更されました。最大解像度は 1080p で、解像度が 480p の場合のビットレートは最大 1.5 Mbps、解像度が 480～1080p の場合のビットレートは最大 3.5 Mbps にできます。これらのドキュメントの変更を参照してください。

- IVS の開始方法 - 「[初期チャンネルの設定](#)」でスクリーンショットを更新しました。
- ストリーミング設定 - 「[チャンネルタイプ](#)」で定義を更新しました。
- コスト - 「[ライブ動画](#)」でチャンネル定義を更新しました。
- Service Quotas - 「[その他のクォータ](#)」で、BASIC チャンルのインGEST ビットレートとインGEST 解像度の IVS 情報を更新しました。

[Player SDK 1.12.0 のリリース: Web](#)

プレイヤーガイド ([ウェブ](#)、[Video.js 統合](#)、および [JW プレイヤー統合](#)) で、新しいリリースのバージョン番号とアーティファクトのリンクを更新しました。

2022 年 8 月 9 日

[Amazon IVS ドキュメントのランディングページ](#)

[このページ](#)は、Player SDK リファレンスの最新バージョンを参照しています。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[Broadcast SDK: iOS 1.5.1](#)

7 月 28 日リリースのリリースノートを更新し、修正項目 (メモリリーク) を追加しました。

2022 年 8 月 8 日

[Amazon S3 への自動録画](#)

[JSON メタデータファイル](#)に、duration_ms を使用して録音の長さを決定することについて、recording_started_at と recording_ended_at のメモを追加しました。

2022 年 8 月 8 日

[Amazon IVS Broadcast SDK: Web](#)

1.0.0 バージョン番号を削除し、今後のリリースのドキュメントはでのみ更新されるという注意を追加することで、このリリースの7月21日のエントリを更新しました(こちらおよびリリースノートを参照) GitHub。

2022年8月4日

[コンソールの手順の明確化](#)

ハンバーガーアイコンをクリックすると、ペインが折りたたまれている場合にのみナビゲーションペインが開きます。これは次の3つの場合に発生します。

2022年8月3日

- [IVS の使用開始](#)—「ステップ 5: ライブストリームの表示」
- [ライブストリームのモニタリング](#)—「ストリームセッションデータへのアクセス」と「ヘルスによるストリームのフィルタリング」

[Broadcast SDK release: iOS 1.5.1](#)

Broadcast SDK ガイド: [iOS](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2022 年 7 月 28 日

[Amazon IVS ドキュメントのランディングページ](#)は、ブロードキャスト SDK リファレンスの最新バージョンを参照しています。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[Amazon IVS Broadcast SDK: Web](#)

Web Broadcast SDK の初回リリース。 [Amazon IVS ドキュメントのランディングページ](#)で「Amazon IVS Broadcast SDK」に一覧表示されているドキュメントを参照してください。

2022 年 7 月 21 日

また、「Amazon IVS の開始方法」の「[Amazon IVS Broadcast SDK によるストリーミング](#)」も更新しました。

重要：今後のリリースでは、ドキュメントは GitHub: <https://aws.github.io/amazon-ivs-web-broadcast/> (ここでは更新されません) でのみ更新されます。

[IVS チャットメトリクス](#)

Amazon による Amazon IVS のモニタリング CloudWatch — IVS Chat にメトリクス (Deliveries) が追加されました。

2022 年 7 月 15 日

2023 年 10 月 12 日の更新:
この CloudWatch ドキュメントは削除され、コンテンツは「[IVS 低レイテンシーストリーミングのモニタリング](#)」に移動されました。

[Player SDK リリース: iOS 1.8.3](#)

「[iOS Player Guide](#)」(iOS プレイヤーガイド) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2022 年 7 月 14 日

[Amazon IVS ドキュメントのランディングページ](#)

は、Player SDK リファレンスの最新バージョンを参照しています。

[推定データ使用のスクリーンショット](#)

[コスト](#)で、「データ使用量の推定」のスクリーンショットが更新されました。「オーディオ」レンディションは提供されなくなりました。

2022 年 6 月 30 日

[プレイヤー SDK 1.11.0 のリリース: Web](#)

2022 年 6 月 28 日

プレイヤーガイド ([ウェブ](#)、[Video.js 統合](#)、および [JW プレイヤー統合](#)) で、新しいリリースのバージョン番号とアーティファクトのリンクを更新しました。

[Amazon IVS ドキュメントのランディングページ](#)で、プレイヤー SDK ウェブリファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「[Amazon IVS リリースノート](#)」も参照してください。

[プレイヤー SDK: ウェブガイド](#)で、「既知の問題と回避策」から、適用されなくなった2つの項目を削除しました。

- Video.js 統合を使って iOS モバイルブラウザで録画したコンテンツを再生すると、再生ボタンが正しく機能しない。
- Google Pixel 4 または 4a モバイルブラウザでライブ配信を再生すると、再生が予期せず停止することがあります。

Broadcast SDK 1.5.0 リリース	<p>ブロードキャスト SDK ガイド: Android、iOS で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。</p> <p>Amazon IVS ドキュメントのランディングページでは、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。</p> <p>このリリースについては、「Amazon IVS リリースノート」も参照してください。</p>	2022 年 6 月 22 日
ストリーミング取り込み設定	<p>ストリームの取り込み: コードブック、RTMPS、およびポート 443 で、用語を明確化: IVS 取り込みサーバーを指定します(パスにポート 443 を含む)。</p>	2022 年 6 月 20 日
Service Quotas	<p>IVS Chat のクォータについては、「ルームあたりの SendMessage リクエスト数」にクォータを追加し、SendMessage リクエストの既存のレートクォータがすべてのルームに適用されることを明確にしました。</p>	2022 年 6 月 14 日

[取り込みサーバー形式](#)

[Amazon IVS の使用開始](#)で、「最終チャンネル作成」のスクリーンショットを更新し、取り込みサーバーの現在の形式 (ポート 443、パス /app/) を表示しました。「OBS Studio によるストリーミング」および「FFmpeg による録画動画のストリーミング」の手順を更新しました。

2022 年 6 月 14 日

[Player SDK 1.10.0 のリリース: Web と Android](#)

プレイヤーガイド ([Web](#)、[Android](#)、[Video.js 統合](#)、および [JW プレイヤー統合](#)) で、新しいリリースのバージョン番号とアーティファクトのリンクを更新しました。

2022 年 5 月 24 日

[Amazon IVS ドキュメントのランディングページ](#)では、Player SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[Service Quotas](#)

GetStreamSession およびのコールレートクォータを追加しました ListStreamSessions。(これらの IVS エンドポイントは、以前、ストリームヘルスの起動時に追加されました。)

2022 年 5 月 16 日

[iOS プレイヤーガイド](#)

「既知の問題と回避策」で、サポートされなくなった iOS 10 に関する箇条書きを削除しました。

2022 年 5 月 10 日

- iOS 10 デバイスは、バックグラウンドから戻ったときにクラッシュすることがあります。

回避方法: バックグラウンドに入る前に、レイヤーの player プロパティを nil にセットします。

[Broadcast SDK: カスタム画像ソース](#)

サンプル iOS アプリに新しい CIFilter 実装の箇条書きを追加しました。

2022 年 5 月 10 日

[Web プレイヤーガイド](#)

「コンテンツセキュリティポリシー」で、サードパーティ CDN (*.akamaized.net および *.ext.cloudfront.live.hls.ttvnw.net)からの動画ストリームのドメインを追加しました。

2022 年 4 月 29 日

[Video.js プレイヤーガイド](#)

「イベント」で、許容される event 値のリストから MetadataEventType (使用できなくなった値) を削除しました。

2022 年 4 月 29 日

[セキュリティポリシーの更新](#)

ID [ベースのポリシーの例](#)で、はコンソールポリシー (チャット、Lambda、Amazon CloudWatch) とその概要テキストを変更しました。

2022 年 4 月 29 日

[プライベートチャンネル](#)

[再生トークンの生成と署名](#)

2022 年 4 月 29 日

[名](#)で、トークンスキーマのペイロードフィールドの exp タイムスタンプ値が UTC であることを指定しました。

[OBS Studio のセットアップ](#)

IVS の使用開始 - [OBS Studio](#)

2022 年 4 月 29 日

[によるストリーミング](#)で、サーバーとストリームキーの指定方法を明確にし、動画の解像度、ビットレート、およびキーフレーム間隔を設定する手順を追加しました。

ストリームヘルスの更新

Amazon IVS Live Stream

Health のモニタリング – 「コンソールの手順」で、高解像度 CloudWatch メトリクスのグラフがストリームセッションの詳細ページで利用可能であることを記載しました。「ヘルスによるストリームのフィルタリング」に「のCloudWatch ヘルスディメンション」を追加しました ConcurrentStreams。

Amazon による Amazon IVS のモニタリング CloudWatch — チャネルのヘルスで結果をフィルタリングするために、新しいディメンション (Health) が ConcurrentStreams メトリクスに追加されました。

2023 年 10 月 12 日の更新:
この CloudWatch ドキュメントは削除され、コンテンツは「IVS 低レイテンシーストリーミングのモニタリング」に移動されました。

2022 年 4 月 28 日

[Amazon IVS Chat](#)

2022 年 4 月 26 日

この新機能の初回リリース。新規および更新された情報には、[Amazon IVS ドキュメントのランディングページ](#)からアクセスできます。

- [Amazon IVS Chat の使用開始](#) — 新しいページ (Amazon IVS Chat ユーザーガイド内)。
- [チャットメッセージレビューハンドラ](#) — 新しいページ (Amazon IVS Chat ユーザーガイド内)。
- Amazon による Amazon IVS のモニタリング CloudWatch -- チャット用の新しいメトリクスと新しい名前空間を追加しました。

2023 年 10 月 12 日の更新:
この CloudWatch ドキュメントは削除され、コンテンツは「[IVS 低レイテンシーストリーミングのモニタリング](#)」に移動されました。

2023 年 12 月 28 日の更新:
チャット関連の CloudWatch コンテンツは「[Amazon IVS Chat のモニタリング](#)」に移動されました。

- [セキュリティ](#) — 「データ保護」で、チャット箇条書きを追加しました。「ID およびアクセス管理」で、「Amazon IVS Chat のリ

ソースベースのポリシー」に関するセクションを追加しました。「インフラストラクチャセキュリティ」で、「Amazon IVS Chat」に関するセクションを追加しました。

- [Service Quotas](#) — 「Service Quota の増加」で、調整可能なクォータが更新されました。2つのセクションを「その他のクォータ」に統合しました。「API コールレートクォータ」、「その他のクォータ」、「Service Quotas と使用状況メトリクスの統合」にチャット情報を追加しました。

CloudWatch

- [Amazon IVS ドキュメントのランディングページ](#)に、2つの API リファレンスドキュメントを含む Amazon IVS Chat のセクションを追加しました。「[IVS チャット API ドキュメントの変更](#)」(このページの新しいセクション)を参照してください。

2023 年 12 月 28 日更新:
チャット関連情報を新しい IVS Chat ユーザーガイドに移動しました。その他のドキュメントの変更については、「[ドキュメント履歴](#)」

[\(チャット\)](#)」を参照してください。

[iOS Player 1.8.2 リリース](#)

「[iOS Player Guide](#)」(iOS プレイヤーガイド)で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2022 年 4 月 22 日

リポジトリの [Amazon IVS ドキュメントのランディングページ](#)では、iOS プレイヤー SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

[手動 SDK インストール](#)

「[Broadcast SDK: Android](#)」(ブロードキャスト SDK: Android)の「Getting Started」(開始方法) > 「Install the Library」(ライブラリをインストールする)のセクション、および「[Player: Android Guide](#)」(プレイヤー: Android ガイド)に、手動でのインストールに関する文を追加しました。

2022 年 4 月 19 日

[Broadcast SDK 1.4.0 のリリース](#)

ブロードキャスト SDK ガイド: [Android](#)、[iOS](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2022 年 4 月 19 日

「[Broadcast SDK: Custom Image Sources](#)」(ブロードキャスト SDK: カスタム画像ソース) に新しいページを追加しました。

[Amazon IVS ドキュメントのランディングページ](#)では、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「[Amazon IVS リリースノート](#)」も参照してください。

[iOS プレイヤー 1.8.1 リリース](#)

「[iOS Player Guide](#)」(iOS プレイヤーガイド) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2022 年 3 月 31 日

リポジトリの [Amazon IVS ドキュメントのランディングページ](#)では、iOS プレイヤー SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

Android プレイヤーのデバイスサポート	[Android プレイヤーガイド] で、サポートされているネイティブ Android デバイス (携帯電話とタブレット) が明確になりました。 [プレイヤーの概要] で、 [ネイティブプラットフォーム] セクションに [サポートされるデバイス] テーブル列を追加しました。	2022 年 3 月 23 日
Amazon IVS EventBridge での Amazon の使用	セッション終了イベントを修正し、その説明を更新しました。また、「セッション作成済み」および「ストリーム終了」のイベント説明も明確化しました。	2022 年 3 月 18 日
プレイヤー Video.js 統合	「スクリプトタグによるセットアップ」のステップ 1 で、例に <code></script></code> の終了を追加しました。	2022 年 3 月 4 日

[Broadcast SDK 1.3.0 のリリース](#)

ブロードキャスト SDK ガイド: [Android](#)、[iOS](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2022 年 3 月 3 日

[Amazon IVS ドキュメントのランディングページ](#)

は、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[プレイヤー 1.8.0 のリリース](#)

すべてのプレイヤーガイド ([Web](#)、[Android](#)、[iOS](#)、[Video.js 統合](#)、および [JW Player 統合](#)) で、新しいリリースのバージョン番号とアーティファクトリンクを更新しました。

2022 年 3 月 1 日

[Amazon IVS ドキュメントのランディングページ](#)

は、Player SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[Amazon IVS EventBridge での Amazon の使用](#)

Recording End Failure イベントに関して、マスタープレイリストの書き込みの試行に失敗するという失敗ケースの例を追加しました。

2022 年 2 月 10 日

[Amazon IVS EventBridge での Amazon の使用](#)

Recording Start イベントに関して、マニフェストファイルと動画セグメントが書き込まれるまでに時間がかかるという注記を追加しました。

2022 年 2 月 9 日

[Broadcast SDK: Android 1.2.1 のリリース](#)

ブロードキャスト SDK ガイド: [Android](#) で、新しいリリースのバージョン番号とアーティファクトのリンクが更新されました。

2022 年 2 月 3 日

リポジトリの [Amazon IVS ドキュメントのランディングページ](#)では、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを参照するようになりました。

このリリースについては、「[Amazon IVS リリースノート](#)」も参照してください。

[Player SDK 用の React Native ラッパー](#)

Player [Android ガイド](#)および [Player iOS ガイド](#)に、新しい React Native ラッパーの GitHub コードとドキュメントへのリンクを追加しました。

2022 年 1 月 27 日

Player SDK 用の React Native ラッパー	Player Android ガイド および Player iOS ガイド に、新しい React Native ラッパーの GitHub コードとドキュメントへのリンクを追加しました。	2022 年 1 月 27 日
Web プレーヤーの CSP の変更	「別のオリジンでのアセットのホスティング」に Chrome の情報を追加しました。	2022 年 1 月 25 日
プライベートチャンネルの設定	「トークンスキーマ」に、access-control-allow-origin トークンペイロードフィールドでの複数のドメインとワイルドカードドメインのサポートに関する情報を追加しました。	2022 年 1 月 24 日
Web プレーヤー 1.7.0 リリースノート	リリースノート の setInitialBufferDuration() に関する箇条書き項目を、iOS モバイルブラウザでは機能しないとなるように更新しました。	2022 年 1 月 21 日

[プレイヤー 1.7.0 のリリース](#)

すべてのプレイヤーガイド ([Web](#)、[Android](#)、[iOS](#)、[Video.js](#) [統合](#)、および [JW Player 統合](#)) で、新しいリリースのバージョン番号とアーティファクトリンクを更新しました。

2022 年 1 月 20 日

[Amazon IVS ドキュメント](#) [のランディングページ](#)で

は、Player SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「[Amazon IVS リリースノート](#)」も参照してください。

[R2S3 サムネイル設定のリリース](#)

Amazon IVS の使用開始の [ステップ 3: 任意の録画によるチャンネルの作成](#) を更新しました。

2022 年 1 月 18 日

[Amazon S3 への自動録画](#) で、「録画の内容」に thumbnails フォルダの変更に関するメモを追加、新しい「サムネイル」セクションを追加、および「JSON メタデータファイル」の thumbnails と path フィールドに関する情報を変更しました。

[Android プレイヤーガイド](#)

JCenter が非推奨になったため、「ライブラリのインストール」の jcenter() 行を削除しました。

2022 年 1 月 7 日

[iOS プレイヤー](#)

arm64e アーキテクチャに対するテスト時にプレイヤーがクラッシュすることに関する「既知の問題」を追加しました。

2021 年 12 月 20 日

[Broadcast SDK 1.2.0 のリリース](#)

すべてのブロードキャスト SDK ガイドで、新しいリリースのバージョン番号と Artifact のリンクが更新されました:
[Android](#)、[iOS](#)。

2021 年 12 月 9 日

リポジトリの [Amazon IVS ドキュメントのランディングページ](#)では、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを参照するようになりました。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[Amazon IVS EventBridge での Amazon の使用](#)

ストリーミング/セッションの開始/作成/終了イベントの説明を充実させ、使用方法の提案を追加しました。

2021 年 12 月 3 日

[ストリーミング設定](#)

Android と iOS からのストリーミングについて、Larix Broadcaster の情報を Amazon IVS Broadcast SDK のドキュメントへのポインタに置き換えました。

2021 年 11 月 24 日

[ブロードキャスト: SDK for Android のガイド](#)

Android 5/6/7 デバイスで
使用できるのがシステムの
デフォルトのマイクのみで
あるため、マイクに対する
ブロードキャスト SDK の
onDeviceAdded コールバッ
クと onDeviceRemoved
コールバックを受信できない
という問題を追加しました。

2021 年 11 月 24 日

[プレイヤー 1.6 のリリース](#)

すべてのプレーヤーガイド
([Web](#)、[Android](#)、[iOS](#)、[Video.js](#)
[統合](#)、および [JW Player 統
合](#)) で、新しいリリースのバー
ジョン番号とアーティファク
トリンクを更新しました。

2021 年 11 月 23 日

[Amazon IVS ドキュメント
のランディングページ](#)
では、Player SDK リファレンス
のリンクが更新され、新しい
バージョンを表示するよう
になりました。

このリリースについては、
「Amazon IVS [リリースノー
ト](#)」も参照してください。

[Amazon IVS Player](#)

導入テキストの末尾に、
キャストのサポート
に関する段落と Amazon IVS
Broadcast SDK ドキュメン
トへのポインタを追加しまし
た。

2021 年 11 月 23 日

[Amazon IVS ライブストリームのヘルスのモニタリング](#)

Amazon IVS の新機能に関する新しいユーザーガイドページストリームのヘルスについては、次の更新もあります。

2021 年 11 月 18 日

- [Amazon Interactive Video Service の使用開始](#)の「ステップ 2: IAM アクセス許可の設定」にある IAM ポリシーを更新し、3 つの IVS 許可 (GetStream 、 GetStreamSession 、 ListStreamSessions) と cloudwatch:GetMetricData を追加しました。
- Amazon による Amazon IVS のモニタリングに、 CloudWatch IngestAudioBitrate 、 、 IngestFrameRate 、 IngestVideoBitrate の 4 つの高解像度メトリクスを追加しましたKeyframeInterval 。

2023 年 10 月 12 日の更新:
この CloudWatch ドキュメントは削除され、コンテンツは「[IVS 低レイテンシーストリーミングのモニタリング](#)」に移動されました。

- 「[Amazon IVS EventBridge での Amazon の使用](#)」に、セッションの作成とセッ

ションの終了の 2 つのイベントを追加しました。

[Amazon IVS EventBridge での Amazon の使用](#)

Recording Start イベントの説明を更新しました。

2021 年 11 月 5 日

[ブロードキャスト: SDK for iOS のガイド](#)

iOS 12 デバイス AirPods に接続するための「既知の問題」を追加します。

2021 年 11 月 4 日

[FFmpeg によるストリーミング](#)

ストリーミング設定で、FFmpeg が (Windows デスクトップだけでなく) 多くの OS/デバイスで使用できることを明らかにし、ウェブカメラについての箇条書きの例の形式を修正しました。

2021 年 11 月 3 日

[Broadcast SDK \(Android およ び iOS \) 1.1.0 のリリース](#)

2021 年 10 月 20 日

ブロードキャスト SDK ガイドで、新しいリリースのバージョン番号と Artifact のリンクが更新されました: [Android](#) および [iOS](#)。Android では、「ブロードキャスト構成の作成」に新しい setPosition 調整があります。iOS では、新しい高度なユースケース (「バックグラウンド動画を使用する」)、 「ブロードキャスト構成の作成」 のスロット位置の変更、および新しい「既知の問題」があります。

リポジトリの [Amazon IVS ドキュメントのランディングページ](#)では、Broadcast SDK リファレンスのリンクが更新され、新しいバージョンを参照するようになりました。

この機能のドキュメントに、新しいページ「[ブロードキャストリング: ミキサーガイド](#)」を追加しました。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

プライベートチャンネルの設定	「トークンスキーマ」で、「ドメイン」ではなく「オリジン」を参照するように access-control-allow-origin 定義が更新されました。	2021 年 10 月 11 日
Android プレイヤー 1.5.1 のリリース	バグ修正のリリースについては、「Amazon IVS リリースノート 」を参照してください。「 Android プレイヤーガイド 」のリンクとテキストのバージョン番号のリファレンスも更新されました。	2021 年 9 月 29 日
プレイヤー 1.5.0 のリリース	すべてのプレーヤーガイド（ Web 、 Android 、 iOS 、 Video.js 統合 、および JW Player 統合 ）で、新しいリリースのバージョン番号と Artifact のリンクを更新しました。 Amazon IVS ドキュメントのランディングページ では、Player SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。 このリリースについては、「Amazon IVS リリースノート 」も参照してください。	2021 年 9 月 28 日
ストリーミング設定	「オーディオ設定」で、最小ビットレート (96 Kbps) を指定しました。	2021 年 9 月 22 日

Amazon IVS の開始方法	「ステップ 4: ストリーミングソフトウェアの設定」で、30 秒間データが送信されなかった場合の接続解除に関する注意を追加しました。	2021 年 9 月 20 日
アイデンティティベースのポリシーの例	Amazon IVS のセキュリティで、「 Amazon IVS チャンネルへのアクセス 」の例のタイプミスを修正し、末尾に句読文字 (}]) を追加しました。	2021 年 9 月 17 日
プレイヤー 1.4.1 および 1.4.0 の SDK サイズ	プレイヤー 1.4.1 と 1.4.0 のリリースノートで、モバイル SDK サイズの表を修正しました。	2021 年 9 月 16 日
プレイヤー 1.4.1 のリリース	バグ修正のリリースについては、「 Amazon IVS リリースノート 」を参照してください。すべてのプレイヤーガイド (Web 、 Android 、 iOS 、 Video.js 統合 、および JW Player 統合) のバージョン番号と Artifact のリンクも更新しました。 Amazon IVS ドキュメントのランディングページ では、Player SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。 「ストリーミング設定」で、 クローズドキャプション の情報を更新しました。	2021 年 9 月 8 日

[ブロードキャスト: SDK for Android のガイド](#)

「プレビュー ImagePreview 用 `imageView` に `video.set` を設定する」で、軽微なテキストの明確化を行いました。「カメラのスイッチ」で、2 つのタイプミスを修正しました。「ブロードキャスト設定の作成」で、現在使用できない `video.set DefaultAspectMode` への参照を削除しました。

2021 年 9 月 1 日

[FFmpeg によるストリーミング設定](#)

動画ファイルをキャプチャするための設定を修正しました。具体的には、`-g 120` を `-force_key_frames expr:gte(t,n_force_d*2)` に変更しました。これにより、ソース入力のフレームレートに関係なく、エンコーダーは 2 秒ごとにキーフレームを挿入します。

2021 年 8 月 23 日

[Amazon IVS Player: SDK for Web のガイド](#)

Pixel 4/4a モバイルブラウザの新しい「既知の問題」を追加しました。

2021 年 8 月 20 日

[Amazon IVS Player: Video.js 統合](#)

「サンプルコード」で、バージョン番号が 7.14.3 に更新されました。7.14.3 以前のバージョンの Video.js には、セキュリティの脆弱性があります。

2021 年 8 月 19 日

[ストリーミング設定](#)

STANDARD チャンネルタイプで、オーディオは 360p 以下のレンディションでのみトランスコードされ、それ以上ではオーディオがパススルーされるという注意を追加しました。

2021 年 8 月 18 日

[Amazon IVS の開始方法](#)

「Step 2: Set Up IAM Permissions」で、既存のユーザーにポリシーをアタッチする手順を追記しました。この新しい手順は、新しいユーザーを作成し、そのユーザーにポリシーをアタッチする手順が古い手順に追加されません。

2021 年 8 月 11 日

[プレイヤー 1.4.0 のリリース](#)

すべてのプレイヤーガイド ([Web](#)、[Android](#)、[iOS](#)、[Video.js 統合](#)、および [JW Player 統合](#)) で、新しいリリースのバージョン番号とアーティファクトのリンクを更新しました。

2021 年 8 月 10 日

[Amazon IVS ドキュメントのランディングページ](#)では、Player SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[Amazon IVS Player: SDK for Web のガイド](#)

「Setup With NPM」に、独自のドメインからプレーヤーの静的アセットをホストすることに関するメモを追加しました。

2021 年 7 月 30 日

[Amazon IVS の開始方法](#)

「Step 2: Set Up IAM Permissions」で、ポリシー情報と手順を更新しました。

2021 年 7 月 29 日

「Step 3: Create a Channel with Optional Recording」に、「Auto-Record to S3」セクションを追加しました (前の段落の置換)。

「Step 4: Set up Streaming Software」に、「Streaming with the Amazon IVS Broadcast SDK」のセクションを追加しました。

[S3 への自動録画](#)

新しいセクション「[Playback of Recorded Content from Private Buckets](#)」を追加しました。また、このページの概要も更新しました。

2021 年 7 月 28 日

[Amazon IVS Broadcast SDK \(Android および iOS\)](#)

ブロードキャスト SDK for Android および SDK for iOS の初期リリース。Amazon IVS [ドキュメントのランディングページ](#)の新しいセクションである「Amazon IVS Broadcast SDK」に一覧表示されているドキュメントを参照してください。

2021 年 7 月 27 日

Amazon IVS Player	デスクトップブラウザ を更新して、Amazon IVS Player 1.3.0 が新しいバージョンの macOS 版 Safari で超低レイテンシーをサポートすることを示しました。	2021 年 7 月 14 日
Amazon IVS Service Quotas	PutMetadata エンドポイントについて、アカウントあたり 155 TPS の制限を追加しました。	2021 年 6 月 29 日
ivs.rocks	Amazon IVS ユーザーガイド ランディングページ に、ivs.rocks へのリンクと簡単な説明を追加しました。	2021 年 6 月 25 日
プレーヤーブラウザとプラットフォームの要件	Amazon IVS Player で、サポートされているブラウザの最新バージョンを一覧表示するサイトへのリンクが追加されました。	2021 年 6 月 25 日
ストリーミング設定	[チャンネルタイプ] で、チャンネルタイプの定義を更新しました。STANDARD チャンネルの解像度は、最大 1080p です。また、BASIC チャンネルの場合は、最大 480p です。(以前の定義は、垂直方向の解像度だけに関するものでした。)	2021 年 6 月 17 日
コスト	コストに関する新しいページを追加しました。	2021 年 6 月 17 日
Amazon IVS Player: SDK for Android のガイド	新しい「アクセス許可」セクションを追加しました。	2021 年 6 月 17 日

[プレーヤーのモバイルブラウザのサポート](#)

[モバイルブラウザ](#)では、Chrome for iPad と Safari for iPad のサポートに関する情報が追加しました。

2021 年 6 月 14 日

[プレイヤー SDK サイズ](#)

新しい「SDK サイズ」セクションを[Android](#) および [iOS](#) プレーヤーSDKガイドに追加しました。

2021 年 6 月 11 日

[Amazon IVS Player: SDK for Web のガイド](#)

iOS モバイルブラウザ (`player.getQualities()` および `player.getLiveLatency()` 呼び出しできる) でコンテンツを再生する際の「既知の問題」2点を追加しました。

2021 年 6 月 9 日

[サポートされているリージョンとサービスエンドポイント](#)

サポートされているリージョンのリストを、[AWS 全般のリファレンスの Amazon IVS ページ](#)へのリンクに置き換えます。このページは、新しいリージョンのサポートが追加されると自動的に更新されず、Amazon CloudWatch で Amazon IVS をモニタリングするページで変更が行われました。

2021 年 6 月 8 日

2023 年 10 月 12 日の更新:
この CloudWatch ドキュメントは削除され、コンテンツは「[IVS 低レイテンシーストリーミングのモニタリング](#)」に移動されました。

Amazon IVS Player の問題	Web 、 Android 、および iOS プレーヤーの「既知の問題と回避策」に、お客様がすべての問題をサポートに報告することを追記しました。Android 11 エミュレータの問題も追加しました。	2021 年 6 月 4 日
Android と iOS プレイヤー 1.3.3 のリリース	バグ修正のリリースについては、「 Amazon IVS リリースノート 」を参照してください。 Android プレイヤーガイド と iOS プレイヤーガイド のリンクとテキストのバージョン番号のリファレンスも更新されました。 Amazon IVS ドキュメントのランディングページ は、常にプレーヤー SDK リファレンスの最新バージョンを参照しています。	2021 年 6 月 1 日
プライベートチャネルの設定	「再生トークンの生成と署名」の手順に関する情報を更新しました。	2021 年 5 月 26 日
グローバルとリージョン別	「Global Solution, Regional Control」を「Getting Started with Amazon IVS」から「 What Is Amazon IVS? 」に移動させました。	2021 年 5 月 21 日
Amazon IVS Player: Video.js 統合	「サンプルコード」では、Cloudflare のバージョン番号が 7.6.6 から 7.11.4 に更新されました。	2021 年 5 月 20 日

Android プレイヤー 1.3.2 のリリース	バグ修正のリリースについては、「 Amazon IVS リリースノート 」を参照してください。「 Android プレイヤーガイド 」のリンクとテキストのバージョン番号のリファレンスも更新されました。	2021 年 5 月 19 日
Amazon IVS Service Quotas	マイナーな言い回しの変更。タグの最大数に関する記述を削除し、API リファレンスに移動しました。	2021 年 5 月 12 日
Amazon IVS リリースノート	Web プレイヤー 1.3.1 のメモを追加しました。1.3.0 NPM パッケージは存在しますが、動作しません。	2021 年 5 月 11 日
Amazon IVS EventBridge での Amazon の使用	関連するすべての例で「サニタイズされた」値になるように <code>stream_id</code> を更新しました。	2021 年 5 月 10 日
Amazon IVS Player: SDK for Web のガイド	iOS モバイルブラウザで録画されたコンテンツを再生する際の <code>player.seekTo()</code> 呼び出しに関する、既知の問題と回避策を追加しました。	2021 年 5 月 10 日
ストリーミング設定	[Encoder Configuration (エンコーダー設定)] ページが [Streaming Configuration (ストリーミング設定)] に変更されました。	2021 年 5 月 6 日

[Amazon IVS EventBridge での Amazon の使用](#)

「Examples: Recording State Change」で、`recording_duration_ms` フィールドを追加し、`recording_s3_key_prefix` フィールドのサンプル値を変更し、`recording_status_reason` フィールドの値を変更しました。

2021 年 5 月 5 日

[プレイヤー 1.3 のリリース](#)

すべてのプレイヤーガイド ([Web](#)、[Android](#)、[iOS](#)、[Video.js](#) [統合](#)、および [JW Player 統合](#)) で、新しいリリースのバージョン番号とアーティファクトのリンクを更新しました。Android 向けに、「ライブラリのインストール」に `mavenCentral()` を追加しました。

2021 年 5 月 5 日

[Amazon IVS ドキュメントのランディングページ](#)では、Player SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

プレイヤー 1.3.0 以降では、iOS 版 Chrome および Safari で時間指定メタデータがサポートされるようになりました。これについては、[IVS プレイヤー SDK の概要](#) (「モバイルブラウザ」の表) および [動画ストリーム内にメタデータを埋め込む](#) (「メタデータの消費」) に記載されています。

このリリースについては、「Amazon IVS [リリースノート](#)」も参照してください。

[Amazon IVS Service Quotas](#)

「Service Quotas Integration with CloudWatch Usage Metrics」という新しいセクションを追加しました。

2021 年 4 月 26 日

ストリームの最大継続時間	Amazon IVS の開始方法 (「ステップ 4: ストリーミングソフトウェアのセットアップ」) では、ストリームの最大継続時間 (48 時間) についての注意を追加しました。	2021 年 4 月 23 日
IAM ポリシーの変更	複数の IAM ポリシーが変更されました。 <ul style="list-style-type: none">• Amazon IVS の使用開始 — 「Step 2: Set Up IAM Permissions」に、Service Quotas を追加しました。• Amazon IVS のセキュリティ — 「Amazon IVS コンソールの使用」で、ポリシーの例を簡素化し、Service Quotas を追加しました。	2021 年 4 月 22 日

[新しい CloudWatch メトリクス](#)

新しい CloudWatch メトリクスのリリースに関するドキュメントのさまざまな変更：

2021 年 4 月 13 日

- Amazon による Amazon IVS のモニタリング CloudWatch — 新しいメトリクスとして、同時視聴と同時ストリームが追加されました。

2023 年 10 月 12 日の更新:
この CloudWatch ドキュメントは削除され、コンテンツは「[IVS 低レイテンシーストリーミングのモニタリング](#)」に移動されました。

- [Service Quotas](#) — 新しいメトリクスに一致するように、関連するクォータの名前を更新しました。
- [用語集](#) — 「ビュー」を追加しました。

[S3 への自動録画](#)

Amazon IVS の新機能に関する新しいユーザーガイドページ 動画ストリーム内にメタデータを埋め込む

2021 年 4 月 7 日

- [Amazon IVS の開始方法](#) — R2S3 の IAM ポリシーの情報を追加しました。チャンネルを作成するステップを書き換えました。OBS Studio でローカル録画をオプションで有効化する段落を追加しました。録画の無効化に関する新しいセクション。
- [Amazon IVS EventBridge での Amazon の使用](#) — 録画状態変更イベントを追加しました。
- Amazon による Amazon IVS のモニタリング CloudWatch — RecordedTime メトリクスを追加しました。

2023 年 10 月 12 日の更新:
この CloudWatch ドキュメントは削除され、コンテンツは「[IVS 低レイテンシーストリーミングのモニタリング](#)」に移動されました。

- [Amazon IVS のセキュリティ](#) — 「Amazon IVS のサービスにリンクされたロール (SLR) の使用」に関するセクションを追加しました。

- [Service Quotas](#) — 新しい録画設定エンドポイントに「API コールレートクォータ」を追加し、録画設定に「リソースクォータ」制限を追加しました。

[Amazon IVS ストリーミング設定](#)

「クローズドキャプション」では、プレイヤー SDK が 1 言語のみをサポートし、マルチトラックキャプションの再生をサポートしないことが明らかになりました。

2021 年 3 月 29 日

[グローバルとリージョン別](#)

「[What is Amazon IVS](#)」に、グローバルとリージョナルの違いを明確にするために、「Global Solution, Regional Control」の新たなセクションを追加しました。「[Getting Started with Amazon IVS](#)」で、チャンネルの作成手順で、リージョンの選択について言及しました。

2021 年 3 月 25 日

[EventBridge イベントレイテンシーと IDR/キーフレームエンコーダーの設定](#)

一部の EventBridge イベントにおける IDR/Keyframe ビデオエンコーダ設定とレイテンシーの関係を明確にしました。これにより、次の 2 つのドキュメントが影響を受けません。

2021 年 3 月 25 日

- [「Amazon IVS ストリーミング設定」](#) — 「レイテンシーの低減」の IDR/Keyframe の箇条書きを参照してください。
- [「Amazon IVS EventBridge で Amazon を使用する」](#) — 新しい「ストリーム状態変更イベントのレイテンシーに関する注意」を参照してください。

[Amazon による Amazon IVS のモニタリング CloudWatch](#)

がデータ CloudWatch を保持する期間を明確にしました。

2021 年 3 月 18 日

2023 年 10 月 12 日の更新:
この CloudWatch ドキュメントは削除され、コンテンツは [「IVS 低レイテンシーストリーミングのモニタリング」](#) に移動されました。

[ストリーミング設定](#)

[Audio Settings (オーディオ設定)] で、対応ビットレートを (192 から) 320 Kpbs に変更しました。

2021 年 3 月 15 日

TLS の必須バージョン	<p>TLS (Transport Layer Security) の要件を明確化しました。API コールでは、クライアントは TLS 1.0 以降をサポートする必要がありますが、当社では TLS 1.2 以降をお勧めします。ストリーミング/再生には TLS バージョン 1.2 以降が必須です。</p> <p>2 つのドキュメントでの変更点：ストリーミング設定 (「Stream Ingest: Codecs, RTMPS, and Port 443」) およびセキュリティ (「Infrastructure Security」の各セクション)。</p>	2021 年 3 月 15 日
Amazon IVS Player: SDK for Web のガイド	HTML5 と <code>setQuality()</code> の既知の問題を追加しました。	2021 年 3 月 15 日
Amazon IVS Player: SDK for Web のガイド	キャプションに関する既知の問題を追加しました。	2021 年 3 月 11 日
Amazon IVS Player	<p>SDK for Android および SDK for iOS のガイドに「スレッドの安全性」に関するセクションを追加しました。</p> <p>また、Android では、<code>player.release()</code> メソッドが呼び出されるとプレイヤーは使用できなくなりますのでご注意ください。</p>	2021 年 3 月 2 日

[Amazon による Amazon IVS のモニタリング CloudWatch](#)

CloudWatch コンソールを使用して Amazon IVS メトリクスにアクセスする手順を更新しました。「IVS」が一覧表示されるタイミングに関する情報とスクリーンショットを追加しました。

2021 年 2 月 26 日

2023 年 10 月 12 日の更新:
この CloudWatch ドキュメントは削除され、コンテンツは「[IVS 低レイテンシーストリーミングのモニタリング](#)」に移動されました。

[セキュリティ](#)

「インフラストラクチャセキュリティ」に、Amazon IVS ストリーミングには TLS 1.2 が必要であるという注記が追加されました。また、AWS グローバルネットワークセキュリティ手順の詳細に関する新しいウェブページも記載されました。

2021 年 2 月 17 日

[Amazon IVS Player: JW プレイヤー統合](#)

Amazon IVS プレイヤー向け JW Player プラグインの新しいユーザーガイドページ。JW Player の行を [Web Player ガイド](#) のフレームワークの統合表に追加しました。

2021 年 1 月 28 日

[Amazon IVS EventBridge での Amazon の使用](#)

イベント送信の保証に関する文言を拡充しました。

2021 年 1 月 22 日

Amazon IVS EventBridge での Amazon の使用	追加: イベントがベストエフォートベースで送信されません。	2021 年 1 月 13 日
ストリーミング設定	コーデックのオーディオ設定を AAC から AAC (LC) に変更しました。	2020 年 12 月 18 日
Amazon IVS Service Quotas	「リソースクォータ」に、リソースのタグの最大数を追加しました。	2020 年 12 月 17 日
Android プレイヤー 1.2.1 のリリース	バグ修正のリリースについては、「 Amazon IVS リリースノート 」を参照してください。 Android プレイヤーガイド のリンクとテキストのバージョン番号のリファレンスも更新されました。	2020 年 12 月 16 日
Amazon IVS リリースノート	Amazon IVS Android Player 1.2.0 および 1.1.0 に、SDK がクラッシュする既知の問題を追加しました。	2020 年 12 月 11 日
Amazon IVS の開始方法	再生 URL の箇条書き (「ステップ 3: チャンネルの作成」) に、再生用のカスタムドメインはサポートされていないという注記を追加しました。	2020 年 12 月 4 日
Amazon IVS リリースノート	iOS プレイヤー 1.0.6 および 1.0.0 のダウンロードリンクを削除しました。これらのバージョンは非推奨です。 iOS プレイヤー 1.2.0 に「既知の問題」を追加しました。	2020 年 12 月 4 日

[プレイヤー 1.2.0 のリリース](#)

すべてのプレイヤーガイド ([Web](#)、[Android](#)、[iOS](#)、および [Video.js 統合](#)) で、新しいリリースのバージョン番号とアーティファクトのリンクを更新しました。Android ガイドに既知の問題を追加しました。

2020 年 11 月 23 日

[Amazon IVS ドキュメント](#) [のランディングページ](#)で

は、Player SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

更新済みの「[Amazon IVS リリースノート](#)」も参照してください。

[プライベートチャネルの設定](#)

「再生トークンの生成と署名」のセクションにおいて、JWT ペイロードの `channel-arn` 値は文字列です。

2020 年 11 月 18 日

[Amazon IVS EventBridge での Amazon の使用](#)

stream_id フィールドを多数のイベントに追加しました。これは、チャンネルがライブになるたびに割り当てられる一意のストリーム識別子です。特定のチャンネルで、各ライブストリームに新しい stream_id が付きます。ストリーム ID を使用すると、同じチャンネル上の異なるストリームセッションを区別できます。

2020 年 11 月 12 日

[動画ストリーム内にメタデータを埋め込む](#)

Amazon IVS コンソールで、「時間指定メタデータの表示」に新しいセクションを追加しました。

2020 年 11 月 9 日

[Web プレイヤーガイド](#)

「コンテンツセキュリティポリシー」のセクションを更新しました。特に Safari の使用時に個別のページでアセットをホストする場合に使用します。

2020 年 11 月 4 日

[Service Quotas \(CCV および CCB の制限\)](#)

特に大規模なストリーミングイベントの前に、同時視聴者および同時配信者の適切な制限を確保することの重要性に関する注意事項を追加しました。「[Amazon IVS の開始方法](#)」および「[Amazon IVS Service Quotas](#)」を参照してください。

2020 年 11 月 4 日

Amazon IVS EventBridge での Amazon の使用	Limit Breach イベントの更新: JSON blob の detail セクションでは、これらすべてのイベントで limit_name を使用します (以前は Concurrent Broadcasts のみで表示され、他は limit を表示していました。)	2020 年 10 月 28 日
プライベートチャネルの設定	「再生トークンの生成と署名」のセクションで、JWT ペイロードの exp (有効期限) フィールドは整数です。	2020 年 10 月 27 日
Amazon IVS Service Quotas	チャネル数、同時視聴者、同時配信の 3 つの制限を引き上げました。	2020 年 10 月 27 日
Web プレイヤー 1.1.2 のリリース	バグ修正のリリースについては、「 Amazon IVS リリースノート 」を参照してください。リンクとテキストのバージョン番号のリファレンスは、 Web Player ガイド および Video.js 統合ガイド 内で更新されました。	2020 年 10 月 9 日
解像度クォータとイベントの取り込み	取り込み解像度のサービスクォータと EventBridge イベントを追加しました。「 Amazon IVS Service Quotas 」および「 Amazon IVS EventBridge での Amazon の使用 」を参照してください。	2020 年 10 月 9 日

[プレイヤー 1.1.0 のリリース](#)

すべてのプレイヤーガイド ([Web](#)、[Android](#)、[iOS](#)、および [Video.js 統合](#)) で、新しいリリースのバージョン番号と Artifact のリンクを更新しました。

2020 年 10 月 7 日

iOS ガイドおよび Web ガイドに、「既知の問題」に関する新しいセクションを追加しました。

[Amazon IVS ドキュメントのランディングページ](#)

は、Player SDK リファレンスのリンクが更新され、新しいバージョンを表示するようになりました。

[Amazon IVS Player](#) の概要

で、Android の getSessionId 機能を削除しました (まだ動作しません)。

[プライベートチャネルの設定](#)

新しいセクション「プライベートチャネルのワークフロー」を追加しました。トークンの生成と署名に関するセクションで、ペイロードフィールドの説明と例を明確にしました。再生キーペアのリスト化と取得の例が修正されました。

2020 年 9 月 21 日

[Amazon IVS EventBridge での Amazon の使用](#)

channel_name フィールドが複数のイベントに追加されました。

2020 年 9 月 14 日

動画ストリーム内にメタデータを埋め込む	IAM アクセス許可の設定 (完全な手順とポリシー)、メタデータの挿入 (CLI 手順を追加)、メタデータの使用 (いくつかの GitHub デモにリンク) に関する情報が拡張されました。	2020 年 9 月 14 日
プレイヤーガイド	各プレイヤーの最新バージョンを明確化しました (Web 、 Android 、 iOS 、 Video.js 統合)。	2020 年 9 月 9 日
Amazon IVS の開始方法	新しいストリームをコンソールで表示できるようになるまでに少し時間がかかることを説明しました。	2020 年 9 月 9 日
Amazon IVS リリースノート	Player iOS のダウンロードリンクを、Player iOS Guide の内容と同じになるように変更しました。	2020 年 9 月 9 日
動画ストリーム内にメタデータを埋め込む	関連する AWS ブログ投稿へのリンクを追加しました。	2020 年 9 月 3 日
Amazon IVS Player	プレイヤーの機能の説明を拡大しました。Amazon IVS プレイヤー (サードパーティー製プレイヤーではない) のみの性能が保証されることを明確化しました。	2020 年 9 月 3 日
Amazon IVS Service Quotas	チャンネル、同時視聴者、同時配信のクォータのみを調整できることを示すように修正しました。	2020 年 8 月 31 日

ストリーミング設定	「サードパーティーのストリーミング/転送サービスの回避」に「 レイテンシーの低減 」サブセクションを追加しました。また、VBR よりも CBR を強く推奨する理由の明確化など、複数の変更があります。	2020 年 8 月 24 日
動画ストリーム内にメタデータを埋め込む	Web での 時間指定メタデータの使用 に関する例を更新しました。	2020 年 8 月 24 日
Amazon IVS Player: SDK for Android のガイド	ライブラリのインストール で、コードの例を更新しました。	2020 年 8 月 24 日
Amazon IVS EventBridge での Amazon の使用	「例: Limit Breach」のセクションで、 <code>limit_name</code> 、 <code>limit_value</code> 、 <code>exceeded_by</code> 、 <code>limit_unit</code> などのフィールド名を更新しました。これらの名前には、アンダースコアが含まれます (ダッシュではありません)。	2020 年 8 月 19 日

[プライベートチャンネルの設定](#)

プライベートチャンネルをサポートする Amazon IVS の新機能に関する新しいユーザーガイドページ。これは、複数の既存のドキュメントにも影響します。

2020 年 8 月 19 日

[Amazon IVS の開始方法](#)および [AWS を使用した Amazon IVS API コール](#)のログ記録 [CloudTrail](#): チャンネルに authorized フィールドを追加しました。

[セキュリティ](#): 「権限アクセスと非権限アクセス」に関する新しいセクションを含む複数の変更。

[Service Quotas](#): 再生クォータをいくつか追加しました。

[用語集](#): 再生キーペアを追加しました。

[Amazon IVS の開始方法](#)

[AWS のリージョンごとのサービス](#)に新しいセクションを追加しました。

2020 年 8 月 11 日

[Amazon IVS Player: SDK for iOS のガイド](#)

リファレンスドキュメントとフレームワークのダウンロードへのリンクが 1.0.6 リリースを示すように更新されました。Amazon IVS のリファレンスドキュメントのリンクも更新されました。[ドキュメントのランディングページ](#)

2020 年 8 月 11 日

Amazon IVS EventBridge での Amazon の使用	Amazon IVS EventBridge イベントが Amazon EventBridge コンソールから利用可能になりました。「Amazon IVS の Amazon EventBridge ルールの作成」のセクションを参照してください。	2020 年 8 月 5 日
Amazon IVS Player: Video.js 統合	「NPM でセットアップする」のセクションで、インストールする Video.js npm パッケージへのリンクをバージョン 7.6.6 に更新しました。	2020 年 7 月 30 日
Amazon IVS EventBridge での Amazon の使用	Amazon IVS ストリーム状態およびストリームヘルスの変更の場合、イベント名は event_name というフィールドに入力されます。(eventName ではありません。以前のドキュメントで説明しています)。	2020 年 7 月 29 日
Amazon IVS の開始方法	Amazon IVS の取り込みにポート 443 が必要であることを示すために、ストリーミングソフトウェアのセットアップ手順を変更しました。これは、ストリーミング設定ドキュメントにも影響します。 RTMPS and Port 443 の新規セクションを参照してください。	2020 年 7 月 27 日

Amazon IVS Player: SDK for iOS のガイド	フレームワークを手動でインストールする手順で、最新バージョンのダウンロード場所が変更されました。	2020 年 7 月 27 日
動画ストリーム内にメタデータを埋め込む	時間指定メタデータの使用に関する Android と iOS の例を追加しました。	2020 年 7 月 24 日
新しいサービスとユーザーガイド	こちらは、Amazon Interactive Video Service (IVS) の初リリースです。	2020 年 7 月 15 日

IVS 低レイテンシーストリーミング API リファレンスの変更

API の変更	説明	日付
トークンレス再生の制限	<ul style="list-style-type: none"> 新しいリソースを追加しました PlaybackRestrictionPolicy。 5 つの PlaybackRestrictionPolicy(ies) エンドポイント (作成/削除/取得/リスト/更新) を追加しました。 オブジェクト PlaybackRestrictionPolicy と PlaybackRestrictionPolicySummary オブジェクトを追加しました。 Channel および ChannelSummary オブジェクト playbackRestrictionPolicyArn にを追加しました。これは、チャンネルエンドポイントのレスポンス (作成/取得BatchGet/リスト/更新) に影響します。 ListChannel リクエストで、を追加しました filterByPlaybackRestrictionPolicyArn 。 	2024 年 1 月 31 日

API の変更	説明	日付
チャンネルタイプの定義	チャンネルタイプの定義を更新して、特にレンダリングされたトランスコードラダーに関する詳細が追記されました。「IVS 低レイテンシーストリーミング API リファレンス」の チャンネルタイプ を参照してください。	2023 年 8 月 18 日
R2S3 レンデーションフィルタリングおよびサムネイルの強化	<ul style="list-style-type: none"> • で ThumbnailConfiguration、resolution とを追加しました storage。これは、CreateRecordingConfiguration リクエストとレスポンス、GetRecordingConfiguration レスポンス、レスポンス GetStreamSession に影響します。 • で ThumbnailConfiguration、は targetIntervalSeconds 最小値を 5 から 1 に変更し、「重要」ノートを更新して BASIC、チャンネルにのみ適用されることを記載しました。 • RenditionConfiguration オブジェクトを追加しました。 • オブジェクト renditionConfiguration にを追加しました Recording Configuration。これは、CreateRecordingConfiguration、の 3 つのレスポンスに影響 GetRecordingConfiguration し ます GetStreamSession。また、リクエスト renditionConfiguration にを追加 しました CreateRecordingConfiguration。 	2023 年 7 月 17 日

API の変更	説明	日付
プライベートチャンネルのビューアーセッションの取り消し	<ul style="list-style-type: none"> • との 2 つのエンドポイントを追加 StartViewerSessionRevocation しました BatchStartViewerSessionRevocation。 • との 2 つのオブジェクトを追加 BatchStartViewerSessionRevocationError しました BatchStartViewerSessionRevocationViewerSession。 	2023 年 6 月 28 日
Advanced チャンネルタイプ	<ul style="list-style-type: none"> • 新しいチャンネル type 値と定義を追加しました。これは、2 つのリクエスト (作成/UpdateChannel) と Channel オブジェクトに影響します。 • Channel および ChannelSummary オブジェクトに preset フィールドを追加しました。これは、いくつかのリクエスト (作成/UpdateChannel) とレスポンス (BatchGetChannel、作成/取得/UpdateChannel GetStreamSession、) に影響しません ListChannels。 • ChannelSummary オブジェクト type に を追加しました。これは ListChannels レスポンスに影響します。 	2023 年 6 月 2 日
RTMP サポート	Channel および ChannelSummary オブジェクトに insecureIngest フィールドを追加しました。これはいくつかのリクエストとレスポンスに影響します。	2023 年 3 月 30 日
ストリームの状態	Stream および StreamSummary オブジェクトで、state フィールドの OFFLINE 値は依存すべきではないことに注意してください。代わりに、NotBroadcasting 「」エラーはストリームがライブではないことを示します。	2023 年 2 月 8 日

API の変更	説明	日付
断片化されたストリームの結合	CreateRecordingConfiguration リクエストと RecordingConfiguration オブジェクトに recordingReconnectWindowSeconds フィールドを追加しました。これは 3 つのレスポンス (CreateRecordingConfiguration、GetRecordingConfiguration、および) に影響します GetStreamSession。	2022 年 8 月 30 日
BASIC チャンネルを 1080p に拡張	CreateChannel、UpdateChannel、および Channel オブジェクトのチャンネルtype定義を更新しました。	2022 年 8 月 16 日
tags の制限事項	tagsの制限事項を更新しました。Amazon IVS には、リンク先の AWS ドキュメントに記載されている内容以外に、タグの制約はありません。これは、「Welcome」セクションと、複数のエンドポイントおよびデータ型に影響します。	2022 年 8 月 12 日
maxResults の最大値とデフォルト値	システムの実際の動作を反映するように、maxResults の最大値とデフォルト値を更新しました。maxResults を使用するすべての List エンドポイントに影響します。	2022 年 8 月 12 日
[Timestamp] (タイムスタンプ) フィールド	[ISO 8601] フィールドについて、これらが文字列として返される旨の備考を追記しました。自動生成の問題により、これらは文書化された構文では number として表示されます。	2022 年 3 月 28 日
API 認証	「Authorization versus Authentication」(認証と認可) で、認可に関する箇条書きを明確にします。	2022 年 3 月 18 日

API の変更	説明	日付
タグエンドポイントでの ARN エンコード	3 つのタグエンドポイントに対して、resourceArn フィールドは URL エンコードされている必要があるという記述を追加しました。	2022 年 3 月 18 日
オーディオ/動画設定オブジェクト	AudioConfiguration および VideoConfiguration データ型の定義を更新し、モニタリングに使用されていることを示します。(設定はブロードキャスターのエンコーダーで行います。)	2022 年 2 月 17 日
R2S3 サムネイル設定のリリース	RecordingConfiguration オブジェクトに新しいフィールド (thumbnailConfiguration) を追加しました。これは、リクエストとレスポンス、GetRecordingConfiguration レスポンス、レスポンス GetStreamSession に影響しません CreateRecordingConfiguration。 新しい オブジェクトを追加しました ThumbnailConfiguration。	2022 年 1 月 18 日
Stream Health リリース	2 つのエンドポイント: GetStreamSession および を追加しました ListStreamSessions。 次の 7 つのオブジェクトを追加しました: AudioConfiguration IngestConfiguration StreamEvent、StreamFilters、StreamSession StreamSessionSummary、および VideoConfiguration。 ストリームと StreamSummary オブジェクトに streamID フィールドを追加しました。これは、GetStream および ListStreams レスポンスにも影響します。 ListStreams リクエストに filtersBy フィールドを追加しました。	2021 年 11 月 18 日

API の変更	説明	日付
時間フィールドの書式	ストリームオブジェクトと StreamSummary オブジェクト <code>startTime</code> の説明を更新し、文字列として返される ISO 8601 タイムスタンプであることを追加しました。	2021 年 9 月 21 日
STANDARD チャンネルタイプ	STANDARD チャンネルタイプで、オーディオは 360p 以下のレンディションでのみトランスコードされ、それ以上ではオーディオがパススルーされるという注意を追加しました。	2021 年 8 月 18 日
ListTagsForResource エンドポイント	ページ割り (<code>maxResults</code> リクエストフィールドと <code>nextToken</code> リクエストレスポンスフィールド) のサポートを削除しました。(ページ割りは正常に機能していませんでした。)	2021 年 8 月 13 日
PutMetadata アカウントあたりの TPS 制限	PutMetadata エンドポイントについて、アカウントあたり 155 TPS の制限を追加しました。	2021 年 6 月 29 日
チャンネルタイプの定義	チャンネルタイプの定義を更新しました。STANDARD チャンネルの解像度は、最大 1080p です。また、BASIC チャンネルの場合は、最大 480p です。(以前の定義は、垂直方向の解像度だけに関するものでした。)	2021 年 6 月 17 日
サポートされているリージョンとサービスエンドポイント	サポートされているリージョンのリストを、 AWS 全般のリファレンスの Amazon IVS ページ へのリンクに置き換えます。このページは、新しいリージョンのサポートが追加されると自動的に更新されます。「Welcome」ページに変更を加えました。	2021 年 6 月 8 日
タグ付け	「タグ付け」(「Welcome」セクション)で、リソースに適用できるタグの最大数(50)を追加しました。	2021 年 5 月 12 日

API の変更	説明	日付
新しい CloudWatch メトリクス	ストリームおよび StreamSummary オブジェクト viewerCount の定義を変更しました。	2021 年 4 月 13 日
S3 への自動録画	<ul style="list-style-type: none"> 4 つの録画設定エンドポイント (Create、Delete、Get、List) を追加しました。 4 つのデータ型 (DestinationConfiguration、RecordingConfiguration、RecordingConfigurationSummary、S3DestinationConfiguration) を追加します。 チャンネル、ChannelSummary オブジェクト、チャンネルエンドポイントに RecordingConfigurationArn フィールドを追加しました。 録画設定 ARN でフィルタリング ListChannels するように変更されました。 	2021 年 4 月 7 日
認証と認可	<ul style="list-style-type: none"> 「Authentication versus Authorization」(認証と認可) セクションを追加し、これらの概念の違いを明確化しました。 authorized フィールド (チャンネルデータタイプとチャンネルエンドポイント内) の説明が、「チャンネルがプライベートかどうか (再生認証が有効かどうか)」に変更されました。 	2021 年 3 月 16 日
PutMetadata	最小文字数 (1) を metadata のリクエストフィールドに追加しました。	2021 年 3 月 4 日
チャンネルレイテンシーモード	「Create/UpdateChannel and Channel/ChannelSummary objects」に、latencyMode 値の説明を追加しました。	2020 年 12 月 18 日

API の変更	説明	日付
チャンネルのデフォルト値	<ul style="list-style-type: none"> • [チャンネルデータタイプ] で、authorized のデフォルト値を追加しました。 • 「チャンネルデータ型」および「」で CreateChannel、 のデフォルト値を追加します type。 	2020 年 12 月 17 日
すべてのリストのエンドポイント	maxResults リクエストフィールドのデフォルト値が 50 であることを示しています。	2020 年 12 月 5 日
ストリームと StreamSummary オブジェクト	viewerCount フィールドの説明を変更し、値が -1 の場合、リクエストがタイムアウトしたことを示すようになりました。この場合は、再試行します。	2020 年 11 月 10 日
認証	Sigv4 の署名情報を追加しました。「 ようこそ 」セクションの「認証」を参照してください。	2020 年 10 月 9 日
DeleteChannel & DeleteStreamKey	HTTP レスポンスが 200 から 204 に変更されました。	2020 年 8 月 26 日
DeleteChannel	エラーを回避するために、ライブ中のチャンネルを削除する方法を明確化しました。	2020 年 8 月 20 日
再生承認 (プライベートチャンネル用)	<ul style="list-style-type: none"> • 新しい PlaybackKeyPair エンドポイント • チャンネルと ChannelSummary オブジェクトの新しい authorized フィールド • 新しいオブジェクト PlaybackKeyPair、および PlaybackKeyPairSummary 	2020 年 8 月 19 日
新しいサービスと API リファレンス	こちらは、Amazon Interactive Video Service (IVS) の初回リリースです。	2020 年 7 月 15 日

ステージ API リファレンスの変更

API の変更	説明	日付
IVS リアルタイムストリーミングのローンチ	<p>このリリースに伴い、ドキュメントに大幅な変更が行われました。以前のドキュメントの名前を「IVS 低レイテンシーストリーミング」に変更し、新しい IVS リアルタイムストリーミングドキュメントを公開しました。「IVS ドキュメントのランディングページ」に、リアルタイムストリーミングと低レイテンシーストリーミングの個別のセクションが追加されました。各セクションには、それぞれのユーザーガイドと API リファレンスがあります。</p> <p>Stage API リファレンスは IVS リアルタイムストリーミングドキュメントの一部であり、「IVS リアルタイムストリーミング API リファレンス」に名前が変更されました。この API リファレンスの将来の更新は、ここではなく ドキュメント履歴 (リアルタイムストリーミング) に一覧を記載します。</p>	2023 年 8 月 7 日
ステージヘルス	<ul style="list-style-type: none">、 GetParticipant、 ListParticipants、 GetStageSession の 5 つのエンドポイントを追加しました ListStageSessions ListParticipantEvents。Event、 Participant、 ParticipantSummary、 の 5 StageSession つのオブジェクトを追加しました StageSessionSummary。	2023 年 5 月 11 日
新しい機能	これはステージ API の最初のリリースです。ドキュメントのランディングページに Stage API リファレンス スタイルを追加しました。	2023 年 3 月 23 日

IVS チャット API ドキュメントの変更

API の変更	説明	日付
新しいエラーメッセージ	ConflictException エラーが追加されました UpdateLoggingConfiguration。	2023 年 3 月 17 日
maximumMessageRate PerSecond の最大値	maximumMessageRatePerSecond の最大値を 10 から 100 に変更しました。これは CreateRoom、GetRoom、および UpdateRoom エンドポイントに影響します。	2023 年 1 月 30 日
Event (Subscribe) MessageID フィールド	「Chat Messaging API リファレンス」の「 Event (Subscribe) 」で、非推奨とされるフィールド (MessageID) を Attributes に追加しました。これは、下位互換性を保つために含まれています。	2023 年 1 月 25 日
新しいエラータイプ	DeleteMessage、DisconnectUser、の PendingVerification エラーを追加しました SendEvent。	2022 年 12 月 5 日
WebSocket エラー	Chat Messaging API リファレンス で、2 つの WebSocket エラー (Unauthorized & Forbidden) の説明を更新しました。	2022 年 11 月 18 日
チャットのログ記録	この新機能の初回リリース。次を追加しました。 <ul style="list-style-type: none"> 新しいリソース、LoggingConfiguration 5 つのデータ型: CloudWatchDestinationConfiguration、DestinationConfiguration KinesisDestinationConfiguration、LoggingConfigurationSummary、S3DestinationConfiguration 5 つのエンドポイント: Create/Delete/Get/List/UpdateLoggingConfiguration(s) オブジェクトとルームの RoomSummary リクエストレスポンスの loggingConfigurationIdentifiers フィールド 	2022 年 11 月 17 日

API の変更	説明	日付
CreateChatToken 説明	attributes フィールドの使用に関する新しいガイダンスを含め CreateChatToken、 の説明を更新しました。	2022 年 11 月 17 日
tags の制限事項	tagsの制限事項を更新しました。Amazon IVS Chat には、リンク先の AWS ドキュメントに記載されている内容以外に、タグの制約はありません。これは、「Welcome」セクション、4 つのエンドポイント (CreateRoom、 ListTagsForResource、 UntagResource) TagResource、 および RoomSummary データ型に影響します。	2022 年 8 月 12 日
新しい機能	<p>これは、Amazon IVS Chat の最初のリリースです。ドキュメントのランディングページに、2 つの API リファレンスドキュメントを含む、Amazon IVS Chat のセクションを追加しました。</p> <ul style="list-style-type: none"> • チャット API リファレンス — コントロールプレーン API (HTTPS) • Chat Messaging API リファレンス -- データプレーン API (WebSocket) 	2022 年 4 月 26 日

リリースノート (Low-Latency Streaming)

2024 年 2 月 1 日

Amazon IVS Broadcast SDK: Android 1.14.1、iOS 1.14.1、Web 1.8.0 (低レイテンシーストリーミング)

プラットフォーム	ダウンロードおよび変更
Web Broadcast SDK 1.8.0	リファレンスドキュメント: https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference/ <ul style="list-style-type: none">変更はありません。
Android Broadcast SDK 1.14.1	リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.14.1/android/ <ul style="list-style-type: none">小さなバグ修正と機能向上。
iOS Broadcast SDK 1.14.1	低レイテンシーストリーミング用のダウンロード: https://broadcast.live-video.net/1.14.1/AmazonIVSBroadcast.xcframework.zip リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.14.1/ios/ <ul style="list-style-type: none">iOS 12 での複数の安定性の問題を修正しました。

Broadcast SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ		
arm64-v8a	1.663 MB	4.708 MB		

アーキテクチャ	圧縮サイズ	非圧縮サイズ		
armeabi-v7a	1.482 MB	3.350 MB		
x86_64	1.804 MB	5.246 MB		
x86	1.793 MB	4.973 MB		

Broadcast SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ		
arm64	1.640 MB	4.010 MB		

2024 年 1 月 31 日

トークンレス再生の制限

このリリースでは、再生承認の外部でオリジンの強制とジオフェンシングが可能になります。IVS 低レイテンシーストリーミングユーザーガイドと API リファレンスが更新されました。変更の詳細については、[ドキュメント履歴](#)を参照してください。

2024 年 1 月 25 日

オーディオのみの再生

IVS はオーディオのみの再生を完全にサポートするようになりました。IVS プレイヤーの概要の「[オーディオのみの再生](#)」および IVS Web プレイヤーガイドの「[オーディオのみの再生](#)」を参照してください。

2024 年 1 月 18 日

Amazon IVS Player SDK 1.24.0

プラットフォーム	ダウンロードおよび変更
<p>Web プレイヤー 1.24.0 と Video.js の統合と JW プレイヤーの統合</p>	<p>NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player</p> <p>スクリプトアセット: https://player.live-video.net/1.24.0/amazon-ivs-player.min.js</p> <p>Video.js テクニカルアセット: https://player.live-video.net/1.24.0/amazon-ivs-videojs-tech.min.js</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.24.0/web/</p> <ul style="list-style-type: none">オーディオのみの再生のサポートが追加されました。オーディオのみの品質は <code>setQuality()</code> で手動で選択する必要があります。自動品質モードでは自動的に選択されません。プレイヤーウェブガイドの オーディオのみの再生 を参照してください。
<p>Android プレイヤー 1.24.0</p>	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.24.0/android/</p> <ul style="list-style-type: none">変更なし
<p>iOS プレイヤー 1.24.0</p>	<p>ダウンロード : https://player.live-video.net/1.24.0/AmazonIVSPlayer.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.24.0/ios/</p> <ul style="list-style-type: none">変更なし

Mobile SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.006 MB	2.846 MB
armeabi-v7a	0.88 MB	1.995 MB
x86_64	1.085 MB	2.916 MB
x86	1.058 MB	2.982 MB

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	0.43 MB	1.08 MB

2024 年 1 月 3 日

Amazon IVS Broadcast SDK: Android 1.13.4、iOS 1.13.4、Web 1.7.0 (低レイテンシーストリーミング)

プラットフォーム	ダウンロードおよび変更
Web Broadcast SDK 1.7.0	リファレンスドキュメント: https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference/ <ul style="list-style-type: none">低レイテンシー SDK の変更はありません。
Android Broadcast SDK 1.13.4	リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.13.4/android/ <ul style="list-style-type: none">低レイテンシー SDK の変更はありません。

プラットフォーム	ダウンロードおよび変更
iOS Broadcast SDK 1.13.4	<p>低レイテンシーストリーミング用のダウンロード : https://broadcast.live-video.net/1.13.4/AmazonIVSBroadcast.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.13.4/ios/</p> <ul style="list-style-type: none"> 低レイテンシー SDK の変更はありません。

Broadcast SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ		
arm64-v8a	1.663 MB	4.704 MB		
armeabi-v7a	1.484 MB	3.352 MB		
x86_64	1.804 MB	5.243 MB		
x86	1.795 MB	4.97 MB		

Broadcast SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ		
arm64	1.63 MB	4.01 MB		

2023 年 12 月 4 日

Amazon IVS Broadcast SDK: Android 1.13.2、iOS 1.13.2 (低レイテンシーストリーミング)

プラットフォーム	ダウンロードおよび変更
Android Broadcast SDK 1.13.2	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.13.2/android/</p> <ul style="list-style-type: none"> 低レイテンシー SDK に変更点はありません。
iOS Broadcast SDK 1.13.2	<p>低レイテンシーストリーミング用のダウンロード: https://broadcast.live-video.net/1.13.2/AmazonIVSBroadcast.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.13.2/ios/</p> <ul style="list-style-type: none"> 低レイテンシー SDK に変更点はありません。

Broadcast SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ		
arm64-v8a	1.663 MB	4.704 MB		
armeabi-v7a	1.484 MB	3.352 MB		
x86_64	1.804 MB	5.243 MB		
x86	1.795 MB	4.970 MB		

Broadcast SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ		
arm64	1.63 MB	4.01 MB		

2023 年 11 月 21 日

Amazon IVS Broadcast SDK: Android 1.13.1 (Low-Latency Streaming)

プラットフォーム	ダウンロードおよび変更
Android Broadcast SDK 1.13.1	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.13.1/android/</p> <ul style="list-style-type: none"> 低レイテンシー SDK に変更点はありません。

Broadcast SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ		
arm64-v8a	1.663 MB	4.705 MB		
armeabi-v7a	1.484 MB	3.352 MB		
x86_64	1.804 MB	5.243 MB		
x86	1.795 MB	4.971 MB		

2023 年 11 月 17 日

Amazon IVS Broadcast SDK: Android 1.13.0、iOS 1.13.0 (Low-Latency Streaming)

プラットフォーム	ダウンロードおよび変更
すべてのモバイル (Android と iOS)	<ul style="list-style-type: none"> • RFC 6555 「Happy Eyeballs」の導入と、ブロードキャストの IPv6 を有効または無効にする BroadcastConfiguration.network.useIPv6 設定オプションの追加により、IPv6 サポートを改善しました。
Android Broadcast SDK 1.13.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.13.0/android/</p> <ul style="list-style-type: none"> • リリース後に AudioSource オブジェクトを使用した場合のクラッシュを修正しました。 • パフォーマンスを向上させるため、Surfaceview ベースのプレビューのサポートが追加されました。Session および StageStream の既存の getPreview メソッドは引き続き TextureView のサブクラスを返しますが、これは今後の SDK バージョンで変更される可能性があります。 • アプリケーションが特に TextureView に依存している場合は、変更せずに続行できます。getPreview から getPreviewTextureView に切り替えて、デフォルトの getPreview が返す内容がいつかは変更されることに備えることもできます。 • アプリケーションが特に TextureView を必要としない場合は、CPU とメモリの使用量を抑えるために getPreview

プラットフォーム	ダウンロードおよび変更
	<p>wSurfaceView に切り替えることをお勧めします。</p> <ul style="list-style-type: none"> アプリケーション提供の Android Surface オブジェクトで動作する ImagePreviewSurfaceTarget という新しいタイプのプレビューが SDK に実装されました。これは Android View のサブクラスではないため、柔軟性が向上しています。
<p>iOS Broadcast SDK 1.13.0</p>	<p>低レイテンシーストリーミング用のダウンロード : https://broadcast.live-video.net/1.13.0/AmazonIVSBroadcast.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.13.0/ios/</p> <ul style="list-style-type: none"> このリリースでは変更された点はありません。

Broadcast SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ		
arm64-v8a	1.66 MB	4.70 MB		
armeabi-v7a	1.48 MB	3.35 MB		
x86_64	1.80 MB	5.24 MB		
x86	1.79 MB	4.96 MB		

Broadcast SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	1.63 MB	4.01 MB

2023 年 11 月 14 日

Amazon IVS Player SDK 1.23.0

プラットフォーム	ダウンロードおよび変更
Web プレイヤー 1.23.0 、 Video.js 統合 、および JW Player 統合	<p>NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player</p> <p>スクリプトアセット: https://player.live-video.net/1.23.0/amazon-ivs-player.min.js</p> <p>Video.js テクニカルアセット: https://player.live-video.net/1.23.0/amazon-ivs-videojs-tech.min.js</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.23.0/web/</p> <ul style="list-style-type: none"> iOS Safari での低レイテンシー再生のサポートを追加しました。
Android プレイヤー 1.23.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.23.0/android/</p> <ul style="list-style-type: none"> リファレンスドキュメントを更新し、新しい UI と他の詳細を追加しました。
iOS プレイヤー 1.23.0	<p>ダウンロード : https://player.live-video.net/1.23.0/AmazonIVSPlayer.xcframework.zip</p>

プラットフォーム	ダウンロードおよび変更
	リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.23.0/ios/ <ul style="list-style-type: none"> 変更はありません。

Mobile SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	0.975 MB	2.744 MB
armeabi-v7a	0.853 MB	1.917 MB
x86_64	1.028 MB	2.873 MB
x86	1.055 MB	2.811 MB

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	0.39 MB	0.93 MB

2023 年 10 月 16 日

Amazon IVS Broadcast SDK: Web 1.6.0 (Low-Latency Streaming)

プラットフォーム	ダウンロードおよび変更
Web Broadcast SDK 1.6.0	リファレンスドキュメント: https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference <ul style="list-style-type: none"> 低レイテンシー SDK の変更はありません。

2023 年 10 月 12 日

Amazon IVS Broadcast SDK: Android 1.12.1 (Low-Latency Streaming)

プラットフォーム	ダウンロードおよび変更
Android Broadcast SDK 1.12.1	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.12.1/android/</p> <ul style="list-style-type: none"> BroadcastSession.setListener の呼び出しがエラーになるバグが修正されました。

Broadcast SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ		
arm64-v8a	1.685 MB	5.046 MB		
armeabi-v7a	1.503 MB	3.702 MB		
x86_64	1.826 MB	5.576 MB		
x86	1.822 MB	5.290 MB		

2023 年 10 月 3 日

Amazon IVS Player SDK 1.22.0

プラットフォーム	ダウンロードおよび変更
Web プレイヤー 1.22.0 、 Video.js 統合 、および JW Player 統合	<p>NPM パッケージ: https://www.npmjs.com/package/amazon-ivs-player</p> <p>スクリプトアセット: https://player.live-video.net/1.22.0/amazon-ivs-player.min.js</p>

プラットフォーム	ダウンロードおよび変更
	<p>Video.js テクニカルアセット: https://player.live-video.net/1.22.0/amazon-ivs-videojs-tech.min.js</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.22.0/web/</p> <ul style="list-style-type: none">• SDK バージョン MediaPlayerPackage.getVersion() を取得する静的メソッドを追加しました。
Android プレイヤー 1.22.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.22.0/android/</p> <ul style="list-style-type: none">• ネットワーク中断後の望ましい再生動作を設定する setNetworkRecoveryMode 関数を Player インターフェイスに追加しました。
iOS プレイヤー 1.22.0	<p>ダウンロード : https://player.live-video.net/1.22.0/AmazonIVSPlayer.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.22.0/ios/</p> <ul style="list-style-type: none">• ネットワーク中断後の望ましい再生動作を設定する setNetworkRecoveryMode 関数を IVSPlayer インターフェイスに追加しました。• IVSPlayerLayer の copyDisplayedPixelBuffer メソッドは、プレイヤーが再生されているときにも使用できるようになりました。以前は、プレイヤーがアイドル状態のときのみ呼び出し可能でした。

Mobile SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	0.948 MB	2.676 MB
armeabi-v7a	0.828 MB	1.865 MB
x86_64	1.025 MB	2.741 MB
x86	1.000 MB	2.802 MB

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	0.37 MB	0.89 MB

2023 年 10 月 2 日

コンソール内ストリーミング

IVS コンソールからストリーミングできるようになりました。「Getting Started with Low-Latency Streaming」の「[ステップ 5: ストリーミングソフトウェアの設定](#)」を参照してください。

2023 年 9 月 14 日

Amazon IVS Broadcast SDK: Web 1.5.2 (Low-Latency Streaming)

プラットフォーム	ダウンロードおよび変更
Web Broadcast SDK 1.5.2	リファレンスドキュメント : https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference

2023 年 8 月 23 日

Amazon IVS Broadcast SDK: Web 1.5.1、Android 1.12.0、iOS 1.12.0 (Low-Latency Streaming)

プラットフォーム	ダウンロードおよび変更
Web Broadcast SDK 1.5.1	リファレンスドキュメント: https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference
Android ブロードキャスト SDK 1.12.0	リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.12.0/android/ <ul style="list-style-type: none">「ピアからシャットダウンを受信した後、受信を試みましたが」というメッセージが表示されてブロードキャストが途中で終了するというまれなバグを修正しました。
iOS Broadcast SDK 1.12.0	低レイテンシーストリーミング用のダウンロード: https://broadcast.live-video.net/1.12.0/AmazonIVSBroadcast.xcframework.zip リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.12.0/ios/ <ul style="list-style-type: none"><code>IVSDeviceDiscovery.createAudioSourceWithName</code> の署名を <code>IVSCustomImageSource</code> ではなく、<code>IVSCustomAudioSource</code> を返すように修正しました。

Broadcast SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ		
arm64-v8a	1.685 MB	5.046 MB		
armeabi-v7a	1.503 MB	3.702 MB		
x86_64	1.826 MB	5.576 MB		
x86	1.822 MB	5.290 MB		

Broadcast SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ		
arm64	1.58 MB	3.88 MB		

2023 年 8 月 23 日

Amazon IVS Broadcast SDK: Android 1.7.6 (Low-Latency Streaming)

プラットフォーム	ダウンロードおよび変更
Android Broadcast SDK 1.7.6	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.7.6/android/</p> <ul style="list-style-type: none">「ピアからシャットダウンを受信した後、受信を試みましたが」というメッセージが表示されてブロードキャストが途中で終了するというまれなバグを修正しました。

Broadcast SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.630 MB	4.689 MB
armeabi-v7a	1.520 MB	3.792 MB
x86_64	1.761 MB	4.748 MB
x86	1.825 MB	5.219 MB

2023 年 8 月 22 日

Amazon IVS Player SDK 1.21.0

プラットフォーム	ダウンロードおよび変更
Web プレイヤー 1.21.0 、 Video.js 統合 、および JW Player 統合	NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player スクリプトアセット: https://player.live-video.net/1.21.0/amazon-ivs-player.min.js Video.js テクニカルアセット: https://player.live-video.net/1.21.0/amazon-ivs-videojs-tech.min.js リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.21.0/web/
Android プレイヤー 1.21.0	リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.21.0/android/
iOS プレイヤー 1.21.0	ダウンロード : https://player.live-video.net/1.21.0/AmazonIVSPlayer.xcframework.zip

プラットフォーム	ダウンロードおよび変更
	<p data-bbox="831 212 1463 296">リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.21.0/ios/</p> <ul data-bbox="831 338 1500 516" style="list-style-type: none"> • IVSPlayerLayer クラスの新しい <code>copyDisplayedPixelBuffer</code> メソッドによって最後に表示されたビデオフレームを取得するサポートが追加されました。

Mobile SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	0.942 MB	2.662 MB
armeabi-v7a	0.823 MB	1.853 MB
x86_64	1.020 MB	2.726 MB
x86	0.993 MB	2.788 MB

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	0.36 MB	0.87 MB

2023 年 8 月 7 日

Amazon IVS Broadcast SDK: Web 1.5.0、Android 1.11.0、iOS 1.11.0

プラットフォーム	ダウンロードおよび変更
Web Broadcast SDK 1.5.0	<p>リファレンスドキュメント : https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference</p> <ul style="list-style-type: none"> Safari で、競合状態によりメディアトラックの取得時に定期的にエラーが発生する問題を修正しました。
Android Broadcast SDK 1.11.0	<p>リファレンスドキュメント : https://aws.github.io/amazon-ivs-broadcast-docs/1.11.0/android</p>
iOS Broadcast SDK 1.11.0	<p>低レイテンシーストリーミング用のダウンロード: https://broadcast.live-video.net/1.11.0/AmazonIVSBroadcast.xcframework.zip</p> <p>リファレンスドキュメント : https://aws.github.io/amazon-ivs-broadcast-docs/1.11.0/ios</p>

Broadcast SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.659 MB	4.918 MB
armeabi-v7a	1.482 MB	3.590 MB
x86_64	1.804 MB	5.444 MB
x86	1.795 MB	5.160 MB

Broadcast SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	0.771 MB	1.879 MB

2023 年 7 月 17 日

R2S3 レンディションのフィルタリングおよびサムネイルの強化

IVS で、Amazon S3 への録画時にストリームに対して生成されるレンディションと、サムネイル用に生成される解像度を制御できるようになりました。詳細については、以下を参照してください。

- 「[IVS の開始](#)」 — 「ステップ 4: チャンネルを作成する」 > 「コンソールの説明」で、スクリーンショットと説明を更新しました。
- 「[Amazon S3 への自動録画](#)」 — 「JSON メタデータファイル」に latest_thumbnail を追加し、thumbnail を更新しました。「サムネイル」と「録画のレンディションを探す」に、レンディション解像度の説明を追加しました。
- 「[コスト](#)」 — 「録画したビデオの保存」で、スクリーンショットを更新しました。
- [IVS API リファレンス](#):
 - で ThumbnailConfiguration、resolution と を追加しました storage。これは、CreateRecordingConfiguration リクエストとレスポンス、GetRecordingConfiguration レスポンス、レスポンス GetStreamSession に影響します。
 - では ThumbnailConfiguration、targetIntervalSeconds 最小値を 5 から 1 に変更し、「重要」ノートを更新して BASIC、チャンネルにのみ適用されることを記載しました。
 - RenditionConfiguration オブジェクトを追加しました。
 - RecordingConfiguration オブジェクト renditionConfiguration に を追加しました。これは、CreateRecordingConfiguration、GetRecordingConfiguration、の 3 つのレスポンスに影響します GetStreamSession。また、CreateRecordingConfiguration リクエスト renditionConfiguration に を追加しました。

2023 年 7 月 14 日

Amazon IVS Player SDK 1.20.0

プラットフォーム	ダウンロードおよび変更
<p>Web プレイヤー 1.20.0、Video.js 統合、および JW Player 統合</p>	<p>NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player</p> <p>スクリプトアセット: https://player.live-video.net/1.20.0/amazon-ivs-player.min.js</p> <p>Video.js テクニカルアセット: https://player.live-video.net/1.20.0/amazon-ivs-videojs-tech.min.js</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.20.0/web/</p> <ul style="list-style-type: none">• iOS モバイル ブラウザでライブストリームまたは録画コンテンツを再生するときに、プレイヤー <code>getLiveLatency()</code> の呼び出しが 0 を返す問題を修正しました。(これは Web プレイヤー 1.17.0 で修正されました。)• <code>amazon-ivs-player</code> npm パッケージの種類定義を修正しました。• Web Player SDK リファレンスで、新しいランディングページを追加し、重複するエントリを削除しました。• Video.js バージョン 8 以降のサポートが追加されました。
<p>Android プレイヤー 1.20.0</p>	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.20.0/android/</p>
<p>iOS プレイヤー 1.20.0</p>	<p>ダウンロード : https://player.live-video.net/1.20.0/AmazonIVSPlayer.xcframework.zip</p>

プラットフォーム	ダウンロードおよび変更
	<p data-bbox="831 214 1464 298">リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.20.0/ios/</p> <ul data-bbox="831 340 1490 466" style="list-style-type: none"> • iOS SDK に iOS 12.0 以上が必要になりました。(iOS 11 はサポートされなくなりました。)

Mobile SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.039 MB	2.922 MB
armeabi-v7a	0.909 MB	2.043 MB
x86_64	1.094 MB	3.069 MB
x86	1.126 MB	3.006 MB

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	0.41 MB	0.99 MB

2023 年 7 月 13 日

Amazon IVS Broadcast SDK: Web 1.4.0、Android 1.10.0、および iOS 1.10.0

プラットフォーム	ダウンロードおよび変更
Web Broadcast SDK 1.4.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference</p> <ul style="list-style-type: none"> • SDK がホストアプリケーションで使用するための入力情報が不十分だったバグを修正しました。 • <code>leave()</code> とそれに続く <code>refreshStrategy()</code> の組み合わせで、終了したにもかかわらずメディアが再公開される場合がある不具合を修正しました。 • <code>stageStreamsToPublish</code> が単一のトラック (オーディオまたはビデオ) を返すと、ストラテジーが更新されたときにクリーンな更新が妨げられる可能性があるバグを修正しました。 • ブラウザタブを閉じたときの切断フローを高速化しました。
すべてのモバイル (Android と iOS)	<ul style="list-style-type: none"> • まれに発生するクラッシュを削減することで、ステージの安定性を向上させました。 • 現在のブロードキャストと同じソケット接続を介して文字列を送信できる新しい <code>sendTimedMetadata</code> メソッドを <code>BroadcastSession</code> に追加しました。この文字列にはタイミング情報が添付されており、IVS Player SDK で受信できます。 • 参加者がステージを離れると、Android で <code>onParticipantLeft</code> が呼び出された

プラットフォーム	ダウンロードおよび変更
<p data-bbox="115 388 548 422">Android Broadcast SDK 1.10.0</p>	<p data-bbox="862 212 1503 338">り、iOS で <code>participantDidLeave</code> が呼び出されたりする前に、参加者の公開状態が未公開に更新されるようになりました。</p> <p data-bbox="829 388 1503 470">リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.10.0/android/</p> <ul data-bbox="829 514 1503 995" style="list-style-type: none">• ステージに再接続中にローテーションすると、他の参加者がクロップされて表示されるバグを修正しました。• <code>AudioStageStream</code> デバイスを <code>AudioDevice</code> にキャストできなかった問題を修正しました。• ラピッド <code>background-to-foreground</code> アプリスイッチにより、サブスクライブされたビデオストリームがミュートされる問題を修正しました。

プラットフォーム	ダウンロードおよび変更
iOS Broadcast SDK 1.10.0	<p>ステージなしのダウンロード : https://broadcast.live-video.net/1.10.0/AmazonIVSBroadcast.xcframework.zip</p> <p>ステージ付きダウンロード : https://broadcast.live-video.net/1.10.0/AmazonIVSBroadcast-Stages.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.10.0/ios/</p> <ul style="list-style-type: none"> ブロードキャストセッションが電話で中断されるときに、視聴者の音声が変わる問題を修正しました。回避策 (電話による中断後にブロードキャストセッションを再開すること) は必要なくなりました。 複数のステージが存在せず、すべてのステージでオーディオを再生できない問題を修正します。 突然ネットワークが失われた場合、接続がタイムアウトするのを待たずに、進行中のブロードキャストがすぐに停止するようになりました。

Broadcast SDK サイズ: Android

アーキテクチャ	ステージ機能なしの圧縮サイズ	ステージ機能なしの非圧縮サイズ	ステージ機能付き圧縮サイズ	ステージ機能付きの非圧縮サイズ
arm64-v8a	1.517 MB	4.761 MB	5.324 MB	15.028 MB
armeabi-v7a	1.340 MB	3.433 MB	4.370 MB	9.489 MB
x86_64	1.653 MB	5.003 MB	5.802 MB	15.837 MB

アーキテクチャ	ステージ機能なしの圧縮サイズ	ステージ機能なしの非圧縮サイズ	ステージ機能付き圧縮サイズ	ステージ機能付きの非圧縮サイズ
x86	1.662 MB	5.287 MB	5.621 MB	15.964 MB

Broadcast SDK サイズ: iOS

アーキテクチャ	ステージ機能なしの圧縮サイズ	ステージ機能なしの非圧縮サイズ	ステージ機能付き圧縮サイズ	ステージ機能付きの非圧縮サイズ
arm64	1.56 MB	3.84 MB	5.04 MB	10.85 MB

2023 年 6 月 28 日

プライベートチャンネルのビューアーセッションの取り消し

IVS のユーザーは、認証トークンに関連付けられたビューアーセッションを取り消して、そのトークンを使用した再生を禁止または停止できるようになりました。詳細については、以下を参照してください。

- [プライベートチャンネルの設定](#) — 「トークンスキーマ」セクションを変更し、「ビューアーセッションの取り消し」を追加しました。
- [IVS API リファレンス](#) — 2 つのエンドポイント (`StartViewerSessionRevocation` および `BatchStartViewerSessionRevocation`) と 2 つのオブジェクト (`BatchStartViewerSessionRevocationError` および) を追加しました `BatchStartViewerSessionRevocationViewerSession`。

2023 年 6 月 27 日

Amazon IVS Broadcast SDK: iOS 1.9.1

プラットフォーム	ダウンロードおよび変更
iOS Broadcast SDK 1.9.1	<p>ステージなしのダウンロード : https://broadcast.live-video.net/1.9.1/AmazonIVSBroadcast.xcframework.zip</p> <p>ステージ付きダウンロード : https://broadcast.live-video.net/1.9.1/AmazonIVSBroadcast-Stages.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.9.1/ios/</p> <ul style="list-style-type: none">• iOS 16.5 以降でビデオのビットレートが徐々に低下する問題を解決しました。(1) 自動ビットレートを使用しない場合、(2) B フレームをオフにして自動ビットレートを使用しない場合は約 20 分、<code>IVSVideoConfiguration.maxBitrate</code> に到達してから約 20 分後にネットワーク接続が安定し、B フレームがオフになっていると、ビデオのビットレートが徐々に低下します。 <p>既知の問題: ブロードキャストセッションが電話で中断されるときに、視聴者の音声が行方不明になる場合があります。回避策は、電話による中断後に、ブロードキャストセッションを再開することです。</p>

Broadcast SDK サイズ: iOS

アーキテクチャ	ステージ機能なしの圧縮サイズ	ステージ機能なしの非圧縮サイズ	ステージ機能付き圧縮サイズ	ステージ機能付きの非圧縮サイズ
arm64	1.55 MB	3.77 MB	5.01 MB	10.77 MB

2023 年 6 月 27 日

Amazon IVS Broadcast SDK 1.7.5

プラットフォーム	ダウンロードおよび変更
iOS Broadcast SDK 1.7.5	<p>ダウンロード : https://broadcast.live-video.net/1.7.5/AmazonIVSBroadcast.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.7.5/ios/</p> <ul style="list-style-type: none"> • iOS 16.5 以降でビデオのビットレートが徐々に低下する問題を解決しました。(1) 自動ビットレートを使用しない場合、(2) B フレームをオフにして自動ビットレートを使用しない場合は約 20 分、IVSVideoC onfiguration.maxBitrate に到達してから約 20 分後にネットワーク接続が安定し、B フレームがオフになっていると、ビデオのビットレートが徐々に低下します。

Broadcast SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	1.48 MB	3.43 MB

2023 年 6 月 16 日

Amazon IVS Broadcast SDK: Web 1.3.3

プラットフォーム	ダウンロードおよび変更
Web Broadcast SDK 1.3.3	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference</p> <ul style="list-style-type: none">IVS のマルチホストヘルスをサポートするために使用される、内部分析でのリグレッションを修正しました。

2023 年 6 月 2 日

Advanced チャンネルタイプ

このリリースでは、既存の BASIC および STANDARD タイプに加えて、2 つの新しいチャンネルタイプ (ADVANCED_SD と ADVANCED_HD) が導入されています。再生と録画で許容可能な解像度とビットレートは、チャンネルタイプによって決まります。

- ADVANCED_SD: 動画がトランスコードされます。元の入力から複数の品質が生成され、視聴者のデバイスとネットワーク状況に最適なエクスペリエンスが自動的に提供されます。入力解像度は最大 1080p、ビットレートは最大 8.5 Mbps にすることができ、出力は SD 品質 (480p) が上限になります。オプションのトランスコードプリセットを選択できます (下記参照)。すべてのレンディションの音声が入力としてトランスコードされ、音声のみのレンディションも利用できます。
- ADVANCED_HD: 動画がトランスコードされます。元の入力から複数の品質が生成され、視聴者のデバイスとネットワーク状況に最適なエクスペリエンスが自動的に提供されます。入力解像度は最大 1080p、ビットレートは最大 8.5 Mbps にすることができ、出力は HD 品質 (720p) が上限になります。オプションのトランスコードプリセットを選択できます (下記参照)。すべてのレンディションの音声が入力としてトランスコードされ、音声のみのレンディションも利用できます。

新しいチャンネルタイプ用のオプションのトランスコードプリセットは、利用可能なダウンロード帯域幅とビデオ品質のトレードオフとして、視聴エクスペリエンスを最適化できるようにします。プリセットには、次の 2 つがあります。

- 制約のある帯域幅配信は、各品質レベルに低いビットレートを使用します。ダウンロード帯域幅が狭い、および/またはシンプルな動画コンテンツ (トーキングヘッドスタイルなど) の場合に使用してください。
- 高帯域幅の配信は、各品質レベルに高いビットレートを使用します。ダウンロード帯域幅が広い、および/または複雑な動画コンテンツ (フラッシュ、すばやいシーンチェンジなど) の場合に使用してください。

「[ドキュメント履歴](#)」ページには、IVS ユーザーガイドおよび IVS API リファレンスに関連する変更がリストされています。

2023 年 6 月 1 日

Amazon IVS Broadcast SDK: Android 1.9.0 および iOS 1.9.0

プラットフォーム	ダウンロードおよび変更
すべて	<ul style="list-style-type: none"> • まれに発生するクラッシュを削減することで、ステージの安定性を向上させました。 • 繰り返し発生するネットワーク中断からの自動復旧を強化しました。
Android ブロードキャスト SDK 1.9.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.9.0/android/</p> <ul style="list-style-type: none"> • 一部のデバイスで、ユーザーがデフォルト以外のマイクに切り替えようとするときに、有線ヘッドフォンの取り外しによってエラーが発生するバグを修正しました。 • 一部のデバイスで、ブロードキャストセッション中にマイクの切り替えを行うと、間違ったマイクがアタッチされるバグを修正しました。 • BroadcastSession、Stage、または DeviceDiscovery オブジェクトを作成する前に Presets メソッドを呼び出すときのクラッシュを修正しました。

プラットフォーム	ダウンロードおよび変更
iOS Broadcast SDK 1.9.0	<p>ステージなしのダウンロード : https://broadcast.live-video.net/1.9.0/AmazonIVSBroadcast.xcframework.zip</p> <p>ステージ付きダウンロード : https://broadcast.live-video.net/1.9.0/AmazonIVSBroadcast-Stages.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.9.0/ios/</p> <ul style="list-style-type: none">• ステージに参加したが、ステージにマイクをアタッチしていないという場合、バックグラウンドに入るときの新しいデフォルト動作は自動的にステージから退出するのではなく、ステージに留まることとなります。これにより、バックグラウンドにいる間に、ビューワー限定参加者としてステージを聴くというユースケースが可能になります。• ステージの Bluetooth デバイス接続/切断の処理が改善されました。• <code>setGain</code> を使用してミュートおよびミュート解除するとき、音声大幅に低くなる問題を修正しました。• カメラを <code>IVSBroadcastSession</code> にアタッチすると、カメラが <code>IVSVideoConfiguration</code> の <code>size</code> と <code>targetFrameRate</code> に基づいて、それ自体を設定するようになりました。• iOS SDK に iOS 12.0 以上が必要になりました。(iOS 11 はサポートされなくなりました。)

プラットフォーム	ダウンロードおよび変更
	既知の問題: ブロードキャストセッションが電話で中断されるときに、視聴者の音声が変わる場合があります。回避策は、電話による中断後に、ブロードキャストセッションを再開することです。

Broadcast SDK サイズ: Android

アーキテクチャ	ステージ機能なしの圧縮サイズ	ステージ機能なしの非圧縮サイズ	ステージ機能付き圧縮サイズ	ステージ機能付きの非圧縮サイズ
arm64-v8a	1.638 MB	4.846 MB	5.451 MB	14.778 MB
armeabi-v7a	1.461 MB	3.532 MB	4.506 MB	9.475 MB
x86_64	1.770 MB	5.082 MB	5.753 MB	15.904 MB
x86	1.781 MB	5.366 MB	5.919 MB	15.708 MB

Broadcast SDK サイズ: iOS

Architectur9	ステージ機能なしの圧縮サイズ	ステージ機能なしの非圧縮サイズ	ステージ機能付き圧縮サイズ	ステージ機能付きの非圧縮サイズ
arm64	1.55 MB	3.77 MB	5.00 MB	10.77 MB

2023 年 5 月 23 日

Amazon IVS Player SDK 1.19.0

プラットフォーム	ダウンロードおよび変更
Web プレイヤー 1.19.0 、 Video.js 統合 、および JW Player 統合	<p>NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player</p> <p>スクリプトアセット : https://player.live-video.net/1.19.0/amazon-ivs-player.min.js</p> <p>Video.js テクニカルアセット : https://player.live-video.net/1.19.0/amazon-ivs-videojs-tech.min.js</p> <p>リファレンスドキュメント : https://aws.github.io/amazon-ivs-player-docs/1.19.0/web/</p>
Android プレイヤー 1.19.0	<p>リファレンスドキュメント : https://aws.github.io/amazon-ivs-player-docs/1.19.0/android/</p> <ul style="list-style-type: none"> 切り替えに十分な帯域幅がある場合でも、再バッファ後にプレイヤーが最低品質のままとなる自動品質モードの問題を修正しました。
iOS プレイヤー 1.19.0	<p>ダウンロード : https://player.live-video.net/1.19.0/AmazonIVSPlayer.xcframework.zip</p> <p>リファレンスドキュメント : https://aws.github.io/amazon-ivs-player-docs/1.19.0/ios/</p>

Mobile SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.013 MB	2.866 MB

アーキテクチャ	圧縮サイズ	非圧縮サイズ
armeabi-v7a	0.919 MB	2.272 MB
x86_64	1.084 MB	3.001 MB
x86	1.058 MB	2.702 MB

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	0.41 MB	0.99 MB

2023 年 5 月 16 日

Amazon IVS Broadcast SDK: iOS 1.8.1

プラットフォーム	ダウンロードおよび変更
iOS Broadcast SDK 1.8.1	<p>ステージなしのダウンロード : https://broadcast.live-video.net/1.8.1/AmazonIVSBroadcast.xcframework.zip</p> <p>ステージ付きダウンロード : https://broadcast.live-video.net/1.8.1/AmazonIVSBroadcast-Stages.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.8.1/ios/</p> <ul style="list-style-type: none"> • iOS 16.4.1 の、RTMP (ステージなし) と WebRTC (ステージあり) におけるビットレート低下の問題を修正しました。アプリに回避策を実装している (b フレームを有効に

プラットフォーム	ダウンロードおよび変更
	している) 場合は、この更新をインストールした後にその回避策を削除できます。

Broadcast SDK サイズ: iOS

アーキテクチャ	ステージ機能なしの圧縮サイズ	ステージ機能なしの非圧縮サイズ	ステージ機能付き圧縮サイズ	ステージ機能付きの非圧縮サイズ
arm64	1.53 MB	3.73 MB	5.00 MB	10.73 MB

2023 年 5 月 16 日

Amazon IVS Broadcast SDK 1.7.4

プラットフォーム	ダウンロードおよび変更
iOS Broadcast SDK 1.7.4	<p>ダウンロード : https://broadcast.live-video.net/1.7.4/AmazonIVSBroadcast.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.7.4/ios/</p> <ul style="list-style-type: none"> • iOS 16.4. のビットレート低下の問題を修正しました。アプリに回避策を実装している (b フレームを有効にしている) 場合は、この更新をインストールした後にその回避策を削除できます。

Broadcast SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	1.48 MB	3.40 MB

2023 年 5 月 11 日

複数ホストのヘルス

複数ホストで IVS ステージのヘルスをモニタリングできるようになりました。以下を参照してください。

- [ステージヘルスのモニタリング](#) — これは、「Amazon IVS ユーザーガイド」の新しいページです。
- [Amazon IVS EventBridge での Amazon の使用](#) – 2 つのステージ更新イベントを追加しました。
- [Amazon IVS Service Quotas](#) — 新しいエンドポイントにコールレートクォータが追加されました。
- [IVS ステージ API リファレンス](#) – 5 つのエンドポイント (GetParticipant、ListStageSessions、ListParticipants GetStageSession、ListParticipantEvents) と 5 つのオブジェクト (イベント、参加者 ParticipantSummary、StageSession) を追加しました StageSessionSummary。

2023 年 5 月 1 日

Amazon IVS Web Broadcast SDK 1.3.2

プラットフォーム	ダウンロードおよび変更
Web Broadcast SDK 1.3.2	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference</p> <ul style="list-style-type: none"> • 画面共有をブロードキャストすると、ライブチャンネルの画面が黒くなることがある問題を修正しました。

プラットフォーム	ダウンロードおよび変更
	<ul style="list-style-type: none"> • ステージ参加者をブロードキャストすると、ライブチャンネルの画面が黒くなることがある問題を修正しました。 • 非公開イベントと公開イベントが重複して報告される分析の問題を修正しました。 • RemoteStageStream オブジェクトの <code>getStats</code> が常に更新されない問題を修正しました。 • ステージ参加者をブロードキャストしようとする際の <code>OverconstrainedError</code> を修正しました。 • ステージ戦略 <code>shouldPublishParticipant</code> が <code>true</code> に設定されている場合、サブスクライブのみの参加者は無視されるという機能強化が追加されました。

2023 年 4 月 27 日

ステージ参加者の増加

一度にステージに接続できる参加者の最大数が 12 人から 1,000 人に増えました。一度に最大 12 人の参加者がステージに公開でき、最大 1,000 人が一度にサブスクライブできます。詳細については、「[Amazon IVS ストリームで複数ホストを有効にする](#)」と「[Amazon IVS Service Quotas](#)」を参照してください。

2023 年 4 月 4 日

Amazon IVS Player SDK 1.18.0

プラットフォーム	ダウンロードおよび変更
Web プレイヤー 1.18.0 、 Video.js 統合 および JW プレイヤー統合	NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player

プラットフォーム	<p>ダウンロードおよび変更</p> <p>スクリプトアセット: https://player.live-video.net/1.18.0/amazon-ivs-player.min.js</p> <p>Video.js テクニカルアセット: https://player.live-video.net/1.18.0/amazon-ivs-videojs-tech.min.js</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.18.0/web/</p> <ul style="list-style-type: none"> 更新後、コンソールタブに「HTTP 応答エラー」 - 「読み込み失敗」が表示される Safari の問題を修正しました。
Android プレイヤー 1.18.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.18.0/android/</p> <ul style="list-style-type: none"> 再生レートが 1 倍を超える場合のビデオ再生の問題を修正しました。
iOS Player 1.18.0	<p>ダウンロード : https://player.live-video.net/1.18.0/AmazonIVSPlayer.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.18.0/ios/</p>

Mobile SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.011 MB	2.854 MB
armeabi-v7a	0.916 MB	2.261 MB
x86_64	1.082 MB	2.990 MB
x86	1.055 MB	2.691 MB

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	0.41 MB	0.99 MB

2023 年 3 月 30 日

RTMP サポート

Amazon IVS は、RTMPS に加えて、RTMP (Real-Time Messaging Protocol) ストリーミングをサポートするようになりました。RTMPS は RTMP の安全なバージョンです。RTMP を必要とする特定の検証済みのユースケースがない限り、安全な取り込みには RTMPS を使用することをお勧めします。

RTMP ストリーミングは次の方法で設定できます。

- IVS コンソール — チャネルの初期設定時には [カスタム設定] ボタンを使用し、既存のチャネルを変更する場合は [RTMP インジェストを有効にする] トグルを使用します。
- API – CreateChannel または UpdateChannel リクエストで新しいinsecureIngestフィールドを使用します。「[IVS API リファレンス](#)」を参照してください。

RTMP インジェストエンドポイントの詳細については、「[ストリーミング ソフトウェアのセットアップ](#)」、「[Broadcast Android SDK ガイド](#)」、および「[Broadcast iOS SDK ガイド](#)」を参照してください。

2023 年 4 月 29 日の修正: 「[Broadcast Android SDK ガイド](#)」と「[Broadcast iOS SDK ガイド](#)」を変更して、これらの SDK が RTMPS 取り込みのみをサポートしている (安全ではない RTMP 取り込みはサポートしていない) ことを示しました。

2023 年 3 月 29 日

プライベートチャネル用の使い捨てトークン

「[再生トークンの生成と署名](#)」で、一時使用のトークンを生成するためのオプションフィールド single-use-uuid をペイロードに追加しました。

2023 年 3 月 28 日

Amazon IVS Web Broadcast SDK 1.3.1

プラットフォーム	ダウンロードおよび変更
Web Broadcast SDK 1.3.1	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference</p> <ul style="list-style-type: none">今回のリリースでは、お客様向けの変更はありません。

2023 年 3 月 23 日

ストリーム上の複数ホストのサポート (ステージリソース)

これは新機能の最初のリリースであり、複数の参加者からのビデオを 1 つのライブストリームに結合できるようになりました。ステージは、参加者がリアルタイムでオーディオとビデオを交換できる仮想空間です。その後、ステージをチャンネルにブロードキャストしてより多くの視聴者にリーチしたり、視聴者を「ステージ」に連れて行って進行中の会話に貢献できるアプリケーションを構築したりできます。詳細については、以下を参照してください。

- [IVS ストリームでの複数ホストの有効化](#) (新規ドキュメント)
- [ステージ API リファレンス](#) (新規ドキュメント)
- [Service Quotas](#) (「Amazon IVS ステージ」エンドポイントと [その他のクォータ] > [Amazon IVS] のステージ制限を参照)
- [Amazon IVS Broadcast SDK 1.8.0](#) の同時リリースに関するドキュメントの変更

2023 年 3 月 23 日

Amazon IVS Broadcast SDK: Android 1.8.0、iOS 1.8.0、Web 1.3.0

ストリーム上の複数ホストのサポートが追加されたことに伴い、Android と iOS のブロードキャスト SDK が新しいステージ機能をサポートするように更新されました。

プラットフォーム	ダウンロードおよび変更
すべて	<ul style="list-style-type: none"> • ステージサポートが追加されました。「ストリーム上の複数ホストのサポート (ステージリソース)」を参照してください。 • Broadcast SDK の概要に、ステージプラットフォームの要件を追加しました。
Android Broadcast SDK 1.8.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.8.0/android/</p> <ul style="list-style-type: none"> • gradle ファイルの設定方法 (クラシファイアを使用) に応じて、ステージサポートの有無にかかわらず SDK をダウンロードできます。ステージサポートのない SDK の方が小さくなっています。詳しくは、「Broadcast SDK: Android ガイド」を参照してください。 • ブロードキャスト SDK: Android ガイドに、「Stage SDK による複数のホストの追加」とステージ関連の「既知の問題と回避策」が追加されました。
iOS Broadcast SDK 1.8.0	<p>ステージなしのダウンロード: https://broadcast.live-video.net/1.8.0/AmazonIVSBroadcast.xcframework.zip</p> <p>ステージ付きダウンロード: https://broadcast.live-video.net/1.8.0/AmazonIVSBroadcast-Stages.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.8.0/ios/</p> <ul style="list-style-type: none"> • ポッドファイルの設定方法に応じて、ステージサポートの有無にかかわらず SDK をダウンロードできます。ステージサポートのない SDK の方が小さくなります。インストー

プラットフォーム	ダウンロードおよび変更
	<p>ルの詳細については、「ブロードキャスト SDK: iOS ガイド」を参照してください。</p> <ul style="list-style-type: none"> • ブロードキャスト SDK: iOS ガイドに、「Stage SDK で複数のホストを追加する」とステージ関連の「既知の問題と回避策」を追加しました。 • Apple がビットコードを正式に廃止し、App Store への送信を受け付けなくなったため、SDK からのビットコードのサポートが廃止されました。詳細については、「Xcode 14 リリースノート」を参照してください。
<p>Web Broadcast SDK 1.3.0</p>	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference</p> <ul style="list-style-type: none"> • Broadcast SDK: Web Guide に、「Stage SDKによる複数ホストの追加」とステージ関連の「既知の問題と回避策」を追加しました。

Broadcast SDK サイズ: Android

アーキテクチャ	ステージ機能なしの圧縮サイズ	ステージ機能なしの非圧縮サイズ	ステージ機能付き圧縮サイズ	ステージ機能付きの非圧縮サイズ
arm64-v8a	1.767 MB	5.192 MB	5.886 MB	16.398 MB
armeabi-v7a	1.656 MB	4.263 MB	4.946 MB	10.924 MB
x86_64	1.967 MB	5.735 MB	6.316 MB	17.376 MB
x86	1.894 MB	5.196 MB	6.387 MB	16.730 MB

Broadcast SDK サイズ: iOS

アーキテクチャ	ステージ機能なしの圧縮サイズ	ステージ機能なしの非圧縮サイズ	ステージ機能付き圧縮サイズ	ステージ機能付きの非圧縮サイズ
arm64	1.53 MB	3.73 MB	5.03 MB	10.67 MB

2023 年 3 月 2 日

Amazon IVS Broadcast SDK: Android 1.7.3

プラットフォーム	ダウンロードおよび変更
Android Broadcast SDK 1.7.3	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.7.3/android/</p> <ul style="list-style-type: none"> MediaTek Dimensity 700 SoC を搭載したデバイスでカスタム画像ソースが正しく動作しない問題を修正しました。

Broadcast SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.629 MB	4.688 MB
armeabi-v7a	1.520 MB	3.792 MB
x86_64	1.825 MB	5.218 MB
x86	1.629 MB	4.688 MB

2023 年 2 月 28 日

Amazon IVS Player SDK 1.17.0

プラットフォーム	ダウンロードおよび変更
<p>Web プレイヤー 1.17.0、Video.js 統合 および JW プレイヤー統合</p>	<p>NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player</p> <p>スクリプトアセット: https://player.live-video.net/1.17.0/amazon-ivs-player.min.js</p> <p>Video.js テクニカルアセット: https://player.live-video.net/1.17.0/amazon-ivs-videojs-tech.min.js</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.17.0/web/</p> <ul style="list-style-type: none">モバイル Safari の <code>getLiveLatency</code> メソッドのサポートを実装しました。
<p>Android プレイヤー 1.17.0</p>	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.17.0/android/</p>
<p>iOS Player 1.17.0</p>	<p>ダウンロード : https://player.live-video.net/1.17.0/AmazonIVSPlayer.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.17.0/ios/</p> <ul style="list-style-type: none">Apple がビットコードを正式に廃止し、App Store への送信を受け付けなくなったため、SDK からのビットコードのサポートが廃止されました。詳細については、「Xcode 14 リリースノート」を参照してください。

Mobile SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.009 MB	2.853 MB
armeabi-v7a	0.915 MB	2.260 MB
x86_64	1.081 MB	2.988 MB
x86	1.054 MB	2.690 MB

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	0.41 MB	0.99 MB

2023 年 2 月 16 日

S3 への自動録画用のバイト範囲タグとマニフェストファイル

auto-record-to-S3 機能は、標準の HLS [プレイリストに加えて、バイト範囲](#)プレイリスト生成をサポートするようになりました。詳細については、「[Amazon S3 への自動録画](#)」（「録画の内容」、「バイト範囲プレイリスト」、recording_started と recording_ended の JSON の例の新しい byte_range_playlist フィールド）を参照してください。

2023 年 1 月 31 日

Amazon IVS Chat Client Messaging SDK: Android 1.1.0

プラットフォーム	ダウンロードおよび変更
Android Chat Client Messaging SDK 1.1.0	リファレンスドキュメント: https://aws.github.io/amazon-ivs-chat-messaging-sdk-android/1.1.0/

プラットフォーム	ダウンロードおよび変更
	<ul style="list-style-type: none"> Kotlin コルーチンをサポートするために、新しい IVS Chat Messaging API を <code>com.amazonaws.ivs.chat.messaging.coroutines</code> パッケージに追加しました。新しい Kotlin コルーチンのチュートリアルも参照してください。パート 1 (パートは全部で 2 つあります) は「チャットルーム」です。

Chat Client Messaging SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
すべてのアーキテクチャ (バイトコード)	89 KB	92 KB

2023 年 1 月 17 日

Amazon IVS Player SDK 1.16.0

プラットフォーム	ダウンロードおよび変更
Web プレイヤー 1.16.0 、 Video.js 統合 および JW プレイヤー統合	<p>NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player</p> <p>スクリプトアセット: https://player.live-video.net/1.16.0/amazon-ivs-player.min.js</p> <p>Video.js テクニカルアセット: https://player.live-video.net/1.16.0/amazon-ivs-videojs-tech.min.js</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.16.0/web/</p>

プラットフォーム	ダウンロードおよび変更
	<ul style="list-style-type: none"> SDK のドキュメントを更新して、iOS モバイルブラウザでサポートされていないメソッドを記載しました。
Android プレイヤー 1.16.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.16.0/android/</p> <ul style="list-style-type: none"> 再生リクエストに Origin リクエストヘッダーを含めることができるようにする setOrigin メソッドを追加しました。新しい strict-origin-enforcement フィールドについては、「トークンスキーマ」も参照してください。
iOS Player 1.16.0	<p>ダウンロード : https://player.live-video.net/1.16.0/AmazonIVSPlayer.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.16.0/ios/</p> <ul style="list-style-type: none"> 再生リクエストに Origin リクエストヘッダーを含めることができるようにする setOrigin メソッドを追加しました。新しい strict-origin-enforcement フィールドについては、「トークンスキーマ」も参照してください。

Mobile SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.009 MB	2.852 MB
armeabi-v7a	0.914 MB	2.258 MB
x86_64	1.054 MB	2.689 MB

アーキテクチャ	圧縮サイズ	非圧縮サイズ
x86	1.080 MB	2.987 MB

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	0.41 MB	0.99 MB

2022 年 12 月 9 日

S3 マニフェストファイルへの自動記録に追加されたタイムスタンプ

[Amazon S3 への自動録画](#)を有効にすると、HLS マニフェストファイルが作成されます。これらのファイルには、UTC ISO-8601 形式を使用して、作成時のすべての HLS セグメントのウォールクロック時刻を示す HLS Program-Date-Time (PDT) タグが含まれるようになりました。

2022 年 12 月 6 日

Amazon IVS Broadcast SDK: Android 1.7.2

プラットフォーム	ダウンロードおよび変更
Android Broadcast SDK 1.7.2	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.7.2/android/</p> <ul style="list-style-type: none"> SurfaceSource をサブクラス化するカメラ以外のデバイスによって返される Device.Descriptor が、呼び出しごとに一意の deviceId および urn を提供し、デバイスを識別するためのプロパティが信頼できなくなるバグを修正。 関連付けられたスロットに、追加時 Device.Descriptor.DeviceTyp

プラットフォーム	ダウンロードおよび変更
	<p>e.MICROPHONE の preferred AudioInput 値がある場合、Mixer.get Slots() によるクエリ時に Broadcast Configuration.Mixer.Slot の preferredAudioInput プロパティが null になるバグを修正。</p>

Broadcast SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.628 MB	4.682 MB
armeabi-v7a	1.519 MB	3.786 MB
x86_64	1.701 MB	5.075 MB
x86	1.637 MB	4.605 MB

2022 年 11 月 17 日

チャットのログ記録

これは、新しい機能の最初のリリースです。ログ記録設定を作成して、チャットルームに送信されたメッセージを保存できるようになりました。詳細については、以下を参照してください。

- [チャットのログ記録](#) – 新しいページ。
- [チャットの開始方法](#) – IAM のアクセス許可を更新し、チャットのログ記録を設定する手順を追加しました。
- [Service Quotas](#) – 新しいエンドポイントとログ記録の設定。
- CloudWatch – ログ送信先メトリクスを追加しました。

2023 年 10 月 12 日の更新: この CloudWatch ドキュメントは削除され、コンテンツは [「IVS 低レイテンシーストリーミングのモニタリング」](#) に移動されました。

2023年12月28日の更新: チャット関連の CloudWatch コンテンツは「[Amazon IVS Chat のモニタリング](#)」に移動されました。

- [チャット API リファレンス](#) – LoggingConfiguration リソースと複数のデータ型とエンドポイントを追加しました。詳細については、「[ドキュメント履歴](#)」を参照してください。

2022年11月9日

Amazon IVS Chat Client Messaging SDK: JavaScript 1.0.2

プラットフォーム	ダウンロードおよび変更
JavaScript Chat Client Messaging SDK 1.0.2	<p>リファレンスドキュメント : https://aws.github.io/amazon-ivs-chat-messaging-sdk-js/1.0.2/</p> <ul style="list-style-type: none"> • Firefox に影響する問題を修正しました。クライアントは、エンドポイントを使用して DisconnectUserチャットルームから切断されると、誤ってソケットエラーを受信しました。

2022年11月1日

Amazon IVS Player SDK 1.14.0

プラットフォーム	ダウンロードおよび変更
すべて	<ul style="list-style-type: none"> • まれなクラッシュの発生を減らし、プレイヤーの安定性を向上させました。
Web プレイヤー 1.14.0 、 Video.js 統合 および JW プレイヤー統合	<p>NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player</p> <p>スクリプトアセット: https://player.live-video.net/1.14.0/amazon-ivs-player.min.js</p>

プラットフォーム	ダウンロードおよび変更
	<p>Video.js テクニカルアセット: https://player.live-video.net/1.14.0/amazon-ivs-videojs-tech.min.js</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.14.0/web/</p>
Android プレイヤー 1.14.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.14.0/android/</p> <ul style="list-style-type: none"> • <code>getErrorCode()</code> メソッドを <code>ErrorType</code> クラスに追加しました。
iOS プレイヤー 1.14.0	<p>ダウンロード: https://player.live-video.net/1.14.0/AmazonIVSPlayer.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.14.0/ios/</p> <ul style="list-style-type: none"> • IVS プレイヤー <code>setQuality:adaptive:</code> メソッドを公開しました。

Mobile SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.004 MB	2.840 MB
armeabi-v7a	0.909 MB	2.248 MB
x86_64	1.049 MB	2.678 MB
x86	1.075 MB	2.975 MB

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	0.41 MB	0.99 MB

2022 年 10 月 18 日

Amazon IVS Chat Client Messaging SDK: JavaScript 1.0.1

プラットフォーム	ダウンロードおよび変更
JavaScript Chat Client Messaging SDK 1.0.1	リファレンスドキュメント: https://aws.github.io/amazon-ivs-chat-messaging-sdk-js/1.0.1/

2022 年 10 月 6 日

Amazon IVS Broadcast SDK 1.7.1

プラットフォーム	ダウンロードおよび変更
iOS Broadcast SDK 1.7.1	<p>ダウンロード: https://broadcast.live-video.net/1.7.1/AmazonIVSBroadcast.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.7.1/ios/</p> <ul style="list-style-type: none"> • いくつかのクラスに対して直接リンクするときのリンカーエラーを修正しました。 • ホストアプリケーションでインスタンス化してはいけないクラスの <code>init</code> 関数および <code>new</code> 関数が削除されました。 • SDK により提供されたカメラを使用し、縦横比が 9:16 に設定されているスロットで、対応する 9:16 のカメラ比が正しく使用され

プラットフォーム	ダウンロードおよび変更
	<p>るようになりました。(以前は 3:4 のカメラ比が使用されていました。) FIT アスペクトモードを使用するスロットは、スペース全体を使用するようになりました。(以前はレターボックス化されていました。)</p>

Broadcast SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	1.48 MB	3.40 MB

2022 年 9 月 22 日

Amazon IVS Broadcast SDK 1.7.0

注: 1.6.0 リリースはありません。

プラットフォーム	ダウンロードおよび変更
すべて	<ul style="list-style-type: none"> まれなクラッシュの発生を減らし、安定性を向上させました。 にAutomaticBitrateProfile 列挙型を追加しました VideoConfiguration。これは、ABR アルゴリズムがビデオのビットレートを調整するレートを制御します。 onTransmissionStatsChanged メソッドが追加されました。onBroadcastQualityChanged および onNetworkHealthChanged より詳細な伝送統計情報が含まれています。後者の 2 つの方法は廃止されたため、代わりに onTransmi

プラットフォーム	ダウンロードおよび変更
	<p><code>sessionStatsChanged</code> の使用をお勧めします。</p>
Android Broadcast SDK 1.7.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.7.0/android/</p>
iOS Broadcast SDK 1.7.0	<p>ダウンロード : https://broadcast.live-video.net/1.7.0/AmazonIVSBroadcast.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.7.0/ios/</p> <ul style="list-style-type: none"> IVSBroadcastSessionAudioSessionStrategy.PlayAndRecordDefaultToSpeaker が追加されました。これにより、開発者はハンドセット (iPhone など) を搭載したデバイスがヘッドセットよりもスピーカーを選択するかどうかを指定できます。

Broadcast SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.628 MB	4.682 MB
armeabi-v7a	1.519 MB	3.786 MB
x86_64	1.824 MB	5.212 MB
x86	1.760 MB	4.742 MB

Broadcast SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	1.47 MB	3.40 MB

2022 年 9 月 20 日

Amazon IVS Player SDK 1.13.0

プラットフォーム	ダウンロードおよび変更
ウェブプレイヤー 1.13.0 、 Video.js 統合 および JW プレイヤー統合	<p>NPM パッケージ: https://www.npmjs.com/package/amazon-ivs-player</p> <p>スクリプトアセット: https://player.live-video.net/1.13.0/amazon-ivs-player.min.js</p> <p>Video.js テクニカルアセット: https://player.live-video.net/1.13.0/amazon-ivs-videojs-tech.min.js</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.13.0/web/</p> <ul style="list-style-type: none">• VideoJS seeking() 関数のサポートが追加されました。• 開発の問題の原因となった未使用のタイプ (CaptureEventTypes) を削除しました。• ネットワーク復旧時の断続的な MediaSource エラーを修正しました。 <p>既知の問題: コンソールを開くと、Sawmill Enabled ログが表示されることがあります。ユーザーに影響しないため、この内部ログは非</p>

プラットフォーム	ダウンロードおよび変更
	表示にされるように意図されています。確認したら無視してください。
Android プレイヤー 1.13.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.13.0/android/</p> <ul style="list-style-type: none"> 競合状態に関連する再生のクラッシュを防ぐために、保護をさらに追加しました。 ABR 帯域幅推定の安定性が向上しました。
iOS プレイヤー 1.13.0	<p>ダウンロード : https://player.live-video.net/1.13.0/AmazonIVSPlayer.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.13.0/ios/</p> <ul style="list-style-type: none"> バックグラウンドで再生中に、オーディオのみの再生が停止するバグを修正しました。 競合状態に関連する再生のクラッシュを防ぐために、保護をさらに追加しました。 ABR 帯域幅推定の安定性が向上しました。 SDK リファレンスで、<code>setAutoMaxQuality</code> がビットレートに基づいて品質をフィルタリングすることを明確にしました。 無効な値を無視するように <code>IVSPlayer</code> クラスの <code>setQuality: メソッド</code> を変更しました。

Mobile SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.000 MB	2.829 MB

アーキテクチャ	圧縮サイズ	非圧縮サイズ
armeabi-v7a	0.904 MB	2.237 MB
x86_64	1.070 MB	2.962 MB
x86	1.045 MB	2.665 MB

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	0.44 MB	1.06 MB

2022 年 9 月 15 日

縦型動画の向上 (最終リリース)

本日 Amazon IVS のすべてのお客様向けに、「[縦型動画の向上](#)」で記載されている変更内容の公開を開始しました。変更がすべてのアカウントに反映されるまで 2~3 日かかります。

2022 年 9 月 12 日

Amazon IVS Broadcast SDK 1.5.2: iOS

プラットフォーム	ダウンロードおよび変更
iOS Broadcast SDK 1.5.2	<p>ダウンロード: https://broadcast.live-video.net/1.5.2/AmazonIVSBroadcast.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.5.2/ios/</p> <ul style="list-style-type: none"> ブロードキャスト停止直後に、ブロードキャストのシャットダウンが完了する前にネット

プラットフォーム	ダウンロードおよび変更
	<p>ワーク接続が切断されるという稀なクラッシュを修正しました。</p> <ul style="list-style-type: none"> 致命的なエラーの後に再試行ループがブロードキャストの再開を繰り返し試みる場合のメモリ増加の問題を修正しました。

Broadcast SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	1.39 MB	3.20 MB

2022 年 9 月 8 日

Amazon IVS Chat Client Messaging SDK: Android 1.0.0 および iOS 1.0.0

プラットフォーム	ダウンロードおよび変更
Android Chat Client Messaging SDK 1.0.0	リファレンスドキュメント: https://aws.github.io/amazon-ivs-chat-messaging-sdk-android/1.0.0/
iOS Chat Client Messaging SDK 1.0.0	リファレンスドキュメント: https://aws.github.io/amazon-ivs-chat-messaging-sdk-ios/1.0.0/

Chat Client Messaging SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
すべてのアーキテクチャ (バイトコード)	53 KB	58 KB

Chat Client Messaging SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
ios-arm64_x86_64-simulator (ビットコード)	484 KB	2.4 MB
ios-arm64_x86_64-simulator	484 KB	2.4 MB
ios-arm64 (ビットコード)	1.1 MB	3.1 MB
ios-arm64	233 KB	1.2 MB

2022 年 9 月 2 日

Amazon IVS Web Broadcast SDK 1.2.0

プラットフォーム	ダウンロードおよび変更
Web Broadcast SDK	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference</p> <ul style="list-style-type: none"> • npm パック実行時のバンドルタイプ定義を修正しました。 • 1080 BASIC チャンネルをサポートするプリセット構成を追加しました。

2022 年 8 月 30 日

断片化されたストリームの結合

これは、新しい機能の最初のリリースです。ストリームが Amazon S3 への自動録画に設定されている場合、ストリームが中断されて新しいストリームが開始された際に、Amazon IVS が前のストリームと同じ S3 プレフィックスに録画を試みる期間を指定できるようになりました。つまり、ブロード

キャストが切断され、指定された間隔内で再接続された場合、複数のストリームは1つのブロードキャストと見なされマージされます。詳細については、以下を参照してください。

- Amazon IVS の開始方法 - コンソールと CLI の手順について、「[ステップ 3: 任意の録画によるチャンネルの作成](#)」を更新しました。
- S3 への自動録画 - 新しいセクションである「[断片化されたストリームの結合](#)」を参照してください。
- EventBridge – [例: 記録状態の変化](#) recording_session_id、および recording_session_stream_ids フィールドが追加されました。
- [IVS API リファレンス](#) — CreateRecordingConfiguration リクエストと RecordingConfiguration オブジェクトに recordingReconnectWindowSeconds フィールドを追加しました。これは 3 つのレスポンス (CreateRecordingConfiguration、 GetRecordingConfiguration、) に影響します GetStreamSession。

2022 年 8 月 9 日

Amazon IVS Web Player SDK 1.12.0

プラットフォーム	ダウンロードおよび変更
ウェブプレイヤー 1.12.0 、 Video.js 統合 および JW プレイヤー統合	<p>NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player</p> <p>スクリプトアセット: https://player.live-video.net/1.12.0/amazon-ivs-player.min.js</p> <p>Video.js テクニカルアセット: https://player.live-video.net/1.12.0/amazon-ivs-videojs-tech.min.js</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.12.0/web/</p> <ul style="list-style-type: none"> • 競合状態に関連する再生のクラッシュを防ぐために、保護をさらに追加しました。

2022 年 7 月 28 日

Amazon IVS iOS Broadcast SDK 1.5.1

プラットフォーム	ダウンロードおよび変更
iOS Broadcast SDK 1.5.1	<p>ダウンロード : https://broadcast.live-video.net/1.5.1/AmazonIVSBroadcast.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.5.1/ios/</p> <ul style="list-style-type: none"> 音声のエンコードができず、すべてのブロードキャストが失敗する原因となっていた iOS 16 との互換性の問題を修正しました。この問題は、iOS 向けの IVS Broadcast SDK の以前のすべてのバージョンに影響します。iOS 16 でブロードキャストするには、バージョン 1.5.1 が必要です。 IVSBroadcastSession のイニシャライザに受任者を直接提供する際のメモリリークを修正しました。(回避策は、後で受任者プロパティを設定することでした。)

Broadcast SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	1.36 MB	3.20 MB

2022 年 7 月 21 日

Amazon IVS Web Broadcast SDK

プラットフォーム	ダウンロードおよび変更
Web Broadcast SDK	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-web-broadcast/docs/sdk-reference</p> <ul style="list-style-type: none">これは、Amazon IVS Web Broadcast SDK の初回リリースです。

2022 年 7 月 14 日

Amazon IVS iOS Player SDK 1.8.3

プラットフォーム	ダウンロードおよび変更
iOS プレイヤー 1.8.3	<p>ダウンロード : https://player.live-video.net/1.8.3/AmazonIVSPlayer.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.8.3/ios/</p> <ul style="list-style-type: none">相対パスを含む URL を介して提供された録画コンテンツをプレイヤーが再生できない問題を修正しました。メインスレッドがブロックされたときに発生し得るメモリ増加の問題を修正しました。

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	0.46 MB	1.10 MB

2022 年 6 月 28 日

Amazon IVS Player Web SDK 1.11.0

プラットフォーム	ダウンロードおよび変更
ウェブプレイヤー 1.11.0 、 Video.js 統合 および JW プレイヤー統合	<p>NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player</p> <p>スクリプトアセット: https://player.live-video.net/1.11.0/amazon-ivs-player.min.js</p> <p>Video.js テクニカルアセット: https://player.live-video.net/1.11.0/amazon-ivs-videojs-tech.min.js</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.11.0/web/</p> <ul style="list-style-type: none">• ABR 帯域幅推定の安定性が向上しました。• Video.js 統合を使って iOS モバイルブラウザで録画したコンテンツを再生すると、再生ボタンが正しく機能しない、という問題を修正しました。以前の回避方法 (Video.js を初期化するとき再生ボタンを非表示にする) は必要ではなくなりました。

2022 年 6 月 22 日

Amazon IVS Broadcast SDK 1.5.0

プラットフォーム	ダウンロードおよび変更
すべて	<ul style="list-style-type: none">まれなクラッシュの発生を減らし、安定性を向上させました。高ビットレートストリームの安定性が向上しました。レイテンシーが非常に高いブロードキャストは、エラーコード20401と次のメッセージで終了します。「ネットワークが遅すぎたため、ブロードキャストは終了しました。接続が安定していることを確認するか、ブロードキャストビットレートを下げてください。」これに対するレイテンシーのしきい値は、時間の経過とともに変化する可能性があります。現在は 45 秒です。
Android Broadcast SDK 1.5.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.5.0/android/</p> <ul style="list-style-type: none">新しい設定オプションを Video に追加しました。これにより、ブロードキャストセッション <code>enableTransparency</code> (boolean) および <code>isTransparencyEnabled()</code> の透過性が可能になります。透過設定は、デフォルトでは無効となっています。個々のスロットで <code>fillColor</code> または <code>transparency</code> の値が期待どおりに機能するように <code>Video.enableTransparency</code> を TRUE に設定する必要があることに注意してください。透過性は計算負荷が高くなるため、必要な場合にのみ有効にします。

プラットフォーム	ダウンロードおよび変更
iOS Broadcast SDK 1.5.0	<p>ダウンロード : https://broadcast.live-video.net/1.5.0/AmazonIVSBroadcast.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.5.0/ios/</p> <ul style="list-style-type: none"> IVSReplayKitBroadcastSession を画面共有に使用する場合は、ストリームを適切にシャットダウンするために RPBroadcastSampleHandler::broadcastFinished で IVSReplayKitBroadcastSession::broadcastFinished を呼び出すことをお勧めします。これを行わないと、ストリームがタイムアウトするまでライブのままになる可能性があります。 IVSImagePreviewView は MTKView によってサポートされではなくなりましたが、代わりに AVSampleBufferDisplayLayer ベースの CALayer を持つ通常の UIView があります。

Broadcast SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.539 MB	4.355 MB
armeabi-v7a	1.431 MB	3.483 MB
x86_64	1.729 MB	4.868 MB
x86	1.675 MB	4.436 MB

Broadcast SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	1.36 MB	3.20 MB

2022 年 6 月 9 日

垂直方向の動画の改善

このリリースでは、Amazon IVS が垂直方向の入力 (例: 高さが幅より大きいモバイルデバイスからブロードキャストされるコンテンツ) を処理する方法を改善しました。この変更は、この記事の最後に説明されているように、時間をかけて展開されています。

3 つの変更があります。

- (Standard チャンネルにのみ影響) 垂直入力はコンテンツの幅に基づいてスケーリングされるため、ダウンスケーリングが少なく、視覚的に高画質な出力が得られます。たとえば、この変更が 720 x 1280 入力にどのように影響するかを示します。

名前	以前の幅 x 高さ	新しい幅 x 高さ
1280p	720 x 1280	â€”
720p	404 x 720	720 x 1280
480p	268 x 480	480 x 852
360p	200 x 360	360 x 640
160p	88 x 160	160 x 284

- (Standard チャンネルにのみ影響) 生成されるレンディションは、入力幅以下のレンディションだけです。たとえば、入力が 720 x 1280 の場合、720p、480p、360p、160p のレンディションが得られます。入力幅がレンディションの間にある場合は、入力よりも幅の小さいレンディションがすべて得られます。たとえば、この変更が 540 x 960 入力にどのように影響するかを示します。

名前	以前の幅 x 高さ	新しい幅 x 高さ
960p	540 x 960	â€”
720p	404 x 720	â€”
480p	268 x 480	480 x 852
360p	200 x 360	360 x 640
160p	88 x 160	160 x 284

3. (Standard チャンネルと Basic チャンネルに影響) 垂直入力のレンディションでは、高さではなく幅に基づいたより従来の命名規則が使用されます。たとえば、Basic チャンネルに 360 x 640 を入力した場合、360p という名前の 1 つの出力レンディションがあります。

この名前は、動画プレイリストに NAME 属性として表示され、ユーザー向けの品質セレクター (例) に表示されます。この名前は、記録されたアセットの Amazon S3 ディレクトリ名としても使用されます。たとえば、360 x 640 入力の場合、品質セレクターと Amazon S3 への自動記録ディレクトリ名は 360p60 (以前の値は 640p60) です。

この改善は時間をかけて展開されています。

- 今 — 過去 6 か月間に垂直入力をブロードキャストしましたか？ そうでない場合は、お客様のアカウントでこの変更を有効にしています (具体的には、本日から 1 週間以上)。はいの場合は、AWS Health Dashboard のアカウントイベントセクションにこの変更に関する通知が届きます。
- 2022 年 9 月 15 日 — 残りのすべてのアカウントに対して変更を有効にします。過去 6 か月間に垂直入力をブロードキャストし、この変更をアカウントでより早く有効にする場合は、AWS サポートチケットを送信してください。

重要: 以前の動作に依存するコード (録音の後処理など) がないことを確認してください。たとえば、レンディションの幅/高さがハードコードされたスクリプトがある場合、それを編集する必要があります。そうしないと、この変更が適用された後にスクリプトが壊れる可能性があります。

2022 年 5 月 24 日

Amazon IVS Web および Android Player SDK 1.10.0

プラットフォーム	ダウンロードおよび変更
<p>Web プレイヤー 1.10.0、Video.js 統合、および JW プレイヤー統合</p>	<p>NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player</p> <p>スクリプトアセット: https://player.live-video.net/1.10.0/amazon-ivs-player.min.js</p> <p>Video.js テクニカルアセット: https://player.live-video.net/1.10.0/amazon-ivs-videojs-tech.min.js</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.10.0/web/</p> <ul style="list-style-type: none">• Video.js プラグインの一時停止と再生に関するコンソールエラーを修正しました。• リファレンスドキュメントで、は公開されるべきではない 2 つのタイプ、AutoplayOptions と TypeScript を定義ファイルから削除しましたPlayerEventType.STATE_CHANGED。• setAutoMaxQuality および setAutoMaxVideoSize を使用時にすべての品質が考慮されない問題を修正しました。• 対応するドキュメントとともに、setAutoMaxVideoSize のメソッドを公開しました。• SDK リファレンスで、setAutoMaxQuality がビットレートに基づいて品質をフィルタリングすることを明確にしました。

プラットフォーム	ダウンロードおよび変更
Android プレイヤー 1.10.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.10.0/android/</p> <ul style="list-style-type: none"> • <code>setAutoMaxQuality</code> および <code>setAutoMaxVideoSize</code> を使用時にすべての品質が考慮されない問題を修正しました。 • <code>Player</code> クラスに <code>getVolume()</code> を追加しました。 • SDK リファレンスで、<code>setAutoMaxQuality</code> がビットレートに基づいて品質をフィルタリングすることを明確にしました。 • ウェブプラットフォームの VODs end-of-stream の動作を修正しました。

Mobile SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	0.990 MB	2.805 MB
armeabi-v7a	0.895 MB	2.215 MB
x86_64	1.033 MB	2.643 MB
x86	1.058 MB	2.936 MB

2022 年 4 月 28 日

ストリームヘルスの更新

ライブストリームのヘルスをほぼリアルタイムでモニタリングするための Amazon IVS ストリームヘルスには、2 つの更新があります。

- 高解像度 CloudWatch メトリクスのチャートが、コンソールのストリームセッションの詳細ページで利用できるようになりました。
- ConcurrentStreams のメトリクスに新しいディメンション (Health) を追加し、チャンネルのヘルスで結果をフィルタリングするようにしました。

「Amazon IVS Live Stream Health のモニタリング」と「Amazon による Amazon IVS のモニタリング」を参照してください CloudWatch。

2023 年 10 月 12 日の更新: これらのドキュメントは「[Amazon IVS Low-Latency Streaming のモニタリング](#)」にまとめられました。

2022 年 4 月 26 日

Amazon IVS Chat

これは Amazon IVS Chat の最初のリリースであり、ライブ動画ストリームと一緒に使用できるマネージドライブチャット機能です。新しいドキュメントは、[Amazon IVS ドキュメントのランディングページ](#)からアクセスできます。

- [Amazon IVS Chat の開始方法](#)から始めます。
- 「Amazon IVS Chat ユーザーガイド」にて:
 - 新しいページの「[チャットメッセージレビューハンドラー](#)」を参照してください。
 - Amazon による Amazon IVS のモニタリング、Amazon IVS セキュリティ CloudWatch、Amazon IVS Service Quotas で「チャット」の変更を検索します。

2023 年 10 月 12 日の更新: CloudWatch ドキュメントが削除され、コンテンツは「[IVS 低レイテンシーストリーミングのモニタリング](#)」に移動されました。

2023 年 12 月 28 日更新: すべてのチャット情報が新しい[チャットユーザーガイド](#)にまとめられました。

- ドキュメントランディングページの新しい Amazon IVS Chat セクションには、次の 2 つの API リファレンスがあります。
- [Chat API リファレンス](#) — コントロールプレーン API (HTTPS)。
- [Chat Messaging API リファレンス](#) – データプレーン API (WebSocket) 。

いつものように、ドキュメントの変更は Amazon IVS [ドキュメント履歴](#) に記載されています。

2022 年 4 月 22 日

Amazon IVS iOS Player SDK 1.8.2

プラットフォーム	ダウンロードおよび変更
iOS Player 1.8.2	<p>ダウンロード : https://player.live-video.net/1.8.2/AmazonIVSPlayer.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.8.2/ios/</p> <ul style="list-style-type: none">• iOS 15 以降を実行しているデバイスでのピクチャーインピクチャーのサポートが追加されました。AVPictureInPictureController クラスは、 のインスタンスで直接インスタンス化できます <code>IVSPlayerLayer</code> 。実装例については、公開サンプルアプリケーション を参照してください。• <code>-seekTo:completionHandler:</code> メソッドの完了ハンドラー内から <code>IVSPlayer</code> の状態を操作しているときに発生し得るデッドロックの問題を修正しました。• メインスレッドがブロックされたときに発生し得るメモリ増加の問題を解決するために、1.8.1 リリースで生じた問題を修正しました。

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	0.46 MB	1.10 MB

2022 年 4 月 19 日

Amazon IVS Broadcast SDK 1.4.0

プラットフォーム	ダウンロードおよび変更
すべて	<ul style="list-style-type: none"> まれなクラッシュの発生を減らし、安定性を向上させました。 「Broadcast SDK: Custom Image Sources」(ブロードキャスト SDK: カスタム画像ソース) に新しいページを追加しました。
Android Broadcast SDK 1.4.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.4.0/android/</p> <ul style="list-style-type: none"> Android 12 のターゲティングを有効にするために <code>createServiceNotificationBuilder</code> でバグを修正しました。 ベースライン AVC プロファイルにフォールバックすることで、バグのあるメイン AVC プロファイルを備えたデバイスに関する問題を修正しました。 予期しない例外によるアプリケーションのクラッシュを防ぐために、いくつかのパブリック API メソッド署名にいくつかの NonNull 注釈を追加します。
iOS Broadcast SDK 1.4.0	<p>ダウンロード: https://broadcast.live-video.net/1.4.0/AmazonIVSBroadcast.xcframework.zip</p>

プラットフォーム	ダウンロードおよび変更
	<p data-bbox="831 212 1463 296">リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.4.0/ios/</p> <ul data-bbox="831 338 1503 961" style="list-style-type: none"> • GCD および Darwin 最適化されたロックをより有効に活用し、バッファの再利用を改善することで、SDK 全体で iOS のパフォーマンスを改善しました。 • では BroadcastConfiguration、キーフレーム間隔の最大値を 10 から 5 に変更し、Android との整合性を確保しました。 • 音声エンコーダの質を制御する新しいメソッドを追加しました。IVSAudioConfiguration で、setQuality メソッドを使用します。エンコーダの質を下げると、CPU の使用状況に大きな影響が生じる可能性があります。

Broadcast SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.526 MB	4.324 MB
armeabi-v7a	1.416 MB	3.442 MB
x86_64	1.657 MB	4.393 MB
x86	1.712 MB	4.827 MB

Broadcast SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	1.33 MB	3.13 MB

2022 年 3 月 31 日

Amazon IVS iOS Player SDK 1.8.1

プラットフォーム	ダウンロードおよび変更
iOS プレイヤー 1.8.1	<p>ダウンロード: <非推奨></p> <p>リファレンスドキュメント : https://aws.github.io/amazon-ivs-player-docs/1.8.1/ios/</p> <ul style="list-style-type: none"> • iOS 15 以降を実行しているデバイスでのピクチャーインピクチャーのサポートが追加されました。AVPictureInPictureController クラスは、 のインスタンスで直接インスタンス化できません IVSPlayerLayer 。実装例については、公開サンプルアプリケーションを参照してください。 • メインスレッドがブロックされたときに発生し得るメモリ増加の問題を修正しました。 • -seekTo:completionHandler: メソッドの完了ハンドラー内から IVSPlayer の状態を操作しているときに発生し得るデッドロックの問題を修正しました。

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	0.46 MB	1.10 MB

2022 年 3 月 3 日

Amazon IVS Broadcast SDK 1.3.0

プラットフォーム	ダウンロードおよび変更
すべて	<ul style="list-style-type: none">まれなクラッシュの発生を減らし、安定性を向上させました。32 ビット符号付き整数および 64 ビット浮動小数点 PCM オーディオのサポートが追加されました。
Android Broadcast SDK 1.3.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.3.0/android/</p> <ul style="list-style-type: none">ストリーミング中に Bluetooth ヘッドセットを取り外すとクラッシュする断続的な問題を修正しました。<code>BroadcastSession.onBroadcastQuality</code> メソッドは、初期ブロードキャスト品質の低い値を報告するようになりました。複数の <code>AudioBufferLists</code> を含む PCM バッファのサポートを追加しました。これは USB マイクで一般的です。 <p>Android 1.2.1 リリースからの変更を取り入れています。サーフェスのサイズと回転の変更を適切にサポートするための新しいメソッドとバグ修正:</p> <ul style="list-style-type: none"><code>SurfaceSource.setSize(...)</code> の新しいサイズを設定 <code>SurfaceSource.setSize(...)</code> しないバグを修正しました <code>SurfaceSource</code>。

プラットフォーム	ダウンロードおよび変更
	<ul style="list-style-type: none"> • デバイスのローテーションをラジアンで設定する <code>Device.setRotation(float rotation)</code> メソッドを追加しました。 • 物理ハンドセットのローテーション <code>ImageDevice</code> 時に の自動ローテーションを有効/無効にする <code>ImageDevice.rotateOnConfigurationChanges(boolean enable)</code> メソッドを追加しました。 • 物理ハンドセット <code>ImageDevice</code> がローテーションしたときに が自動的にローテーションするように設定されているかどうかを返す <code>ImageDevice.willRotateOnConfigurationChanges()</code> メソッドを追加しました。
<p>iOS Broadcast SDK 1.3.0</p>	<p>ダウンロード : https://broadcast.live-video.net/1.3.0/AmazonIVSBroadcast.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.3.0/ios/</p> <ul style="list-style-type: none"> • <code>createAppBackgroundImageSource</code> メソッドを使用するときのいくつかの競合状態が修正されました。この競合状態では、アプリがフォアグラウンドに戻った後にストリーミングが再開できない可能性があります。 • arm64 シミュレータのサポートが追加されました。

Broadcast SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.531 MB	4.411 MB
armeabi-v7a	1.420 MB	3.525 MB
x86_64	1.719 MB	4.877 MB
x86	1.659 MB	4.925 MB

Broadcast SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	1.30 MB	3.06 MB

2022 年 3 月 1 日

Amazon IVS Player SDK 1.8.0

プラットフォーム	ダウンロードおよび変更
すべて	<ul style="list-style-type: none"> 録画コンテンツの再生時の品質切り替え時のフリーズの発生が減少しました。
Web プレイヤー 1.8.0 、 Video.js 統合 、および JW プレイヤー統合	<p>NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player</p> <p>スクリプトアセット: https://player.live-video.net/1.8.0/amazon-ivs-player.min.js</p> <p>Video.js テクニカルアセット: https://player.live-video.net/1.8.0/amazon-ivs-videojs-tech.min.js</p>

プラットフォーム	ダウンロードおよび変更
	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.8.0/web/</p> <ul style="list-style-type: none"> 一部のブラウザで録画されたコンテンツの再生が停止する可能性があるエッジケースを修正しました。 録画した動画の転送後に時間指定メタデータイベントがトリガーされない問題を修正しました。 remove() での JW プレイヤー統合に関する不要で混乱する警告を削除しました。 正しいキュータイプフィルタリングをサポートするために、キュータイプの厳密なタイプチェックを有効にしました。
Android プレイヤー 1.8.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.8.0/android/</p> <ul style="list-style-type: none"> ViewUtil クラスを削除。これは内部であり、非推奨でした。代わりに PlayerView を使用します。
iOS プレイヤー 1.8.0	<p>ダウンロード : https://player.live-video.net/1.8.0/AmazonIVSPlayer.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.8.0/ios/</p>

Mobile SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	0.975 MB	2.761 MB
armeabi-v7a	0.882 MB	2.177 MB

アーキテクチャ	圧縮サイズ	非圧縮サイズ
x86_64	1.020 MB	2.603 MB
x86	1.043 MB	2.890 MB

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	0.46 MB	1.10 MB

2022 年 2 月 3 日

Amazon IVS Broadcast SDK: Android 1.2.1

プラットフォーム	ダウンロードおよび変更
Android Broadcast SDK 1.2.1	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.2.1/android/</p> <p>このリリースには、新しいメソッドと、サーフェスのサイズとローテーションの変更を適切にサポートするためのバグ修正が含まれています。これは、カスタム動画入力を伴うユースケースに必要です。</p> <ul style="list-style-type: none"> が の新しいサイズを設定 <code>SurfaceSource.setSize(...)</code> しないバグを修正しました <code>SurfaceSource</code>。 デバイスのローテーションをラジアンで設定する <code>Device.setRotation(float rotation)</code> メソッドを追加しました。 物理ハンドセットのローテーション <code>ImageDevice</code> 時に の自動ローテーションを

プラットフォーム	ダウンロードおよび変更
	<p>有効/無効にする <code>ImageDevice.rotateOnConfigurationChanges(boolean enable)</code> メソッドを追加しました。</p> <ul style="list-style-type: none"> 物理ハンドセット <code>ImageDevice</code> がローテーションしたときに が自動的にローテーションするように設定されているかどうかを返す <code>ImageDevice.willRotateOnConfigurationChanges()</code> メソッドを追加しました。

Broadcast SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.642 MB	4.536 MB
armeabi-v7a	1.468 MB	3.261 MB
x86_64	1.866 MB	5.225 MB
x86	1.809 MB	4.916 MB

2022 年 1 月 20 日

Amazon IVS Player SDK 1.7.0

プラットフォーム	ダウンロードおよび変更
すべて	<ul style="list-style-type: none"> ソースメディアのプレイリストからストリームを再生するときの音の途切れを修正しました。
Web プレーヤー 1.7.0 、 Video.js 統合 、および JW Player 統合	NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player

プラットフォーム	ダウンロードおよび変更
	<p>スクリプトアセット: https://player.live-video.net/1.7.0/amazon-ivs-player.min.js</p> <p>Video.js テクニカルアセット: https://player.live-video.net/1.7.0/amazon-ivs-videojs-tech.min.js</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.7.0/web/</p> <ul style="list-style-type: none">• Amazon IVS で録画した動画の再生後に時間指定メタデータイベントがトリガーされない問題を修正しました。• iOS モバイルウェブブラウザでストリームの再生 URL が使用できない場合に ErrorNotAavailable エラーが発生しない問題を修正しました。• Video.js ラッパーによる dispose() の呼び出し時におけるコンソール警告を修正しました。• プレーヤーインスタンスが破棄された後でのアクセス試行が原因で発生するいくつかの null 参照エラーを修正しました。• setQuality ドキュメントを更新して、成功通知を受け取るには QUALITY_CHANGED をリスンする必要があることを明確にしました。• setInitialBufferDuration() ドキュメントを更新して、iOS モバイルブラウザでは機能しないことを明記しました。 <p>既知の問題: 視聴者が録画されたコンテンツで順方向にスキップしてから逆方向にスキップすると、順方向にスキップした時間が過ぎるまで</p>

プラットフォーム	ダウンロードおよび変更
	<p>iOS ブラウザ内の時間指定メタデータがトリガーされません。例えば、視聴者が録画されたコンテンツの視聴を開始し、60 秒後にスキップしてから 30 秒前にスキップすると、30 秒から 60 秒の間は時間指定メタデータがトリガーされません。この問題は、今後のリリースで修正される予定です。</p>
Android プレイヤー 1.7.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.7.0/android/</p> <ul style="list-style-type: none">• 内部クラスである ViewUtil クラスが非推奨になりました。代わりに PlayerView を使用してください。このクラスは、次回の Amazon IVS Player リリース (1.8.0、2022 年第 1 四半期に提供予定) で完全に削除されます。• 動画がビューに表示される方法を制御する <code>PlayerView.setResizeMode(mode)</code> を追加しました。これは、動画をオプションでズームインしたり、動画のアスペクト比を完全に無視してビュー全体に表示したりすることを可能にします。

プラットフォーム	ダウンロードおよび変更
iOS プレイヤー 1.7.0	<p>ダウンロード : https://player.live-video.net/1.7.0/AmazonIVSPlayer.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.7.0/ios/</p> <ul style="list-style-type: none"> • iOS SDK に iOS 11.0 以上が必要になりました。 • SDK に arm64e スライスが含まれなくなりました。このスライスは、Apple でこれが標準アーキテクチャになり次第、再度有効化されます。 • アプリケーションの終了とメディアサービスのリセットイベント中に発生する可能性があったまれなクラッシュを修正しました。

Mobile SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.013 MB	2.820 MB
armeabi-v7a	0.895 MB	2.012 MB
x86_64	1.119 MB	3.099 MB
x86	1.125 MB	2.970 MB

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	0.46 MB	1.09 MB

2022 年 1 月 18 日

R2S3 サムネイル設定

このリリースでは、ライブセッションのサムネイルの記録を有効/無効にし、ライブセッションのサムネイルが生成される間隔を変更できます。これは、この新しい機能の最初のリリースです。以下を参照してください。

- [Amazon IVS の開始方法](#) – 「ステップ 3: 任意の録画によるチャンネルの作成」が更新されました。
- [Amazon S3 への自動録画](#) – いくつかの変更が行われました。
 - 「録画の内容」に thumbnails フォルダの変更に関するメモを追加。
 - 新しい「サムネイル」セクションを追加。
 - 「JSON メタデータファイル」の thumbnails と path フィールドに関する情報を変更。
- [Amazon IVS API リファレンス](#) – いくつかの変更が行われました。
 - RecordingConfiguration オブジェクトの新しいフィールド (thumbnailConfiguration)。これは、CreateRecordingConfiguration リクエストとレスポンス、GetRecordingConfiguration レスポンス、レスポンス GetStreamSession に影響します。
 - 新しいオブジェクト : ThumbnailConfiguration

2021 年 12 月 9 日

Amazon IVS Broadcast SDK 1.2.0

プラットフォーム	ダウンロードおよび変更
すべて	<ul style="list-style-type: none"> • まれなクラッシュの発生を減らし、安定性を向上させました。 • 新しいメソッド onNetworkHealthChanged (Android) と broadcastSession:networkHealthChanged (iOS) を追加しました。これは、ネットワークの品質が瞬時に変化したときの更新を提供するものです。これは、ブロードキャストに一時的な中断が発生する場合のフィードバックを提供するために使用できます。

プラットフォーム	ダウンロードおよび変更
	<ul style="list-style-type: none">• <code>BroadcastConfiguration.mixer.canvasAspectMode</code> を取得/設定するメソッドを追加しました。これは、スロットのアスペクトモードが明示的に設定されていないときに、スロットのデフォルトのアスペクトモードとして使用されます。• Mixer (Android) API と IVSBroadcastMixer (iOS) API を次のように変更しました。<ul style="list-style-type: none">• 追加されたすべてのスロットを返す <code>getSlots()</code> を追加しました。• ミキサースロットからデバイスをバインド解除する <code>unbind</code> を追加しました。• 成功または失敗を示すブール値を返すように <code>bind</code>、<code>unbind</code>、および <code>transition</code> を更新しました。
Android Broadcast SDK 1.2.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.2.0/android/</p> <ul style="list-style-type: none">• 透過が有効な場合でも、スロットの動画またはイメージがその下にある他のスロットと (<code>zIndex</code> 値を使用して) ブレンドされないバグを修正しました。

プラットフォーム	ダウンロードおよび変更
iOS Broadcast SDK 1.2.0	<p>ダウンロード : https://broadcast.live-video.net/1.2.0/AmazonIVSBroadcast.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.2.0/ios/</p> <ul style="list-style-type: none">• Bluetooth および有線ヘッドセットの接続の安定性が向上しました。• 次のピクセル形式について、IVSCustom ImageSource のサポートを追加しました。<ul style="list-style-type: none">• kCV PixelFormatType_Lossless_420YpCbCr8BiPlanarFullRange• kCV PixelFormatType_Lossy_420YpCbCr8BiPlanarFullRange• kCV PixelFormatType_Lossless_420YpCbCr8BiPlanarVideoRange• kCV PixelFormatType_Lossy_420YpCbCr8BiPlanarVideoRange• kCV PixelFormatType_Lossless_32BGRA• kCV PixelFormatType_Lossy_32BGRA• createAppBackgroundImageSource メソッドを使用するときの 2 つの競合状態が修正されました。この競合状態では、アプリケーションがフォアグラウンドに戻った後にストリーミングが再開できない可能性があります。

Broadcast SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.639 MB	4.530 MB
armeabi-v7a	1.466 MB	3.255 MB
x86_64	1.863 MB	5.219 MB
x86	1.806 MB	4.910 MB

Broadcast SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	1.42 MB	3.30 MB

2021 年 11 月 23 日

Amazon IVS Player SDK 1.6

プラットフォーム	ダウンロードおよび変更
すべて	<ul style="list-style-type: none"> 新しいプレイヤー関数 <code>setInitialBufferDuration()</code> を追加しました。これは、初期バッファ期間を設定できるようにします。この期間は、再生を開始できるタイミングを決定します。許容範囲は 0.1 ~ 5 秒です。このメソッドは、iOS ブラウザプラットフォームには影響しません。 ネットワークの再接続中に <code>play</code> メソッドが呼び出されずに、ロードされたストリームが再生される可能性がある問題を修正しました。

プラットフォーム	ダウンロードおよび変更 <ul style="list-style-type: none">古いクローズドキャプションデータが消去されない問題を修正しました。まれなクラッシュの発生を減らし、プレイヤーの安定性を向上させました。
Web プレイヤー 1.6.1 および Video.js 統合 および JW Player 統合	NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player スクリプトアセット: https://player.live-video.net/1.6.1/amazon-ivs-player.min.js Video.js テクニカルアセット: https://player.live-video.net/1.6.1/amazon-ivs-videojs-tech.min.js リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.6.1/web/ <ul style="list-style-type: none">setQuality のドキュメントに、video 要素の controls 属性が呼び出しに与える影響についてのメモを追加しました。プレイヤーが動画のデコードエラーとプレイリストのネットワークエラーから復旧する方法が向上しました。プレイヤーのデフォルトのログレベルを、他のプラットフォームに合わせて警告からエラーに変更しました。
Android プレイヤー 1.6.0	リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.6.0/android/

プラットフォーム	ダウンロードおよび変更
iOS プレイヤー 1.6.0	<p>ダウンロード : https://player.live-video.net/1.6.0/AmazonIVSPlayer.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.6.0/ios/</p> <ul style="list-style-type: none"> • iOS 10 のサポートは、次回の IVS Player リリース (1.7.0、暫定では 2022年第 1 四半期の予定) から非推奨になる予定です。

Mobile SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.01 MB	2.82 MB
armeabi-v7a	0.84 MB	2.16 MB
x86_64	1.13 MB	2.97 MB
x86	1.12 MB	3.09 MB

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
universal	0.94 MB	2.02 MB
arm64	0.47 MB	1.11 MB
armv7	0.46 MB	0.89 MB

2021 年 11 月 18 日

Stream Health

Amazon IVS Stream Health では、新しい高解像度 CloudWatch メトリクスにより、ライブストリームの状態をほぼリアルタイムでモニタリングできます。また、2 つの新しい API エンドポイントを通してストリームのイベントと入力設定に関する情報を得ることもできます。これは、この新しい機能の最初のリリースです。以下を参照してください。

- [Amazon IVS ライブストリームのヘルスのモニタリング](#) — これは「Amazon IVS ユーザーガイド」の新しいページです。
- [Amazon Interactive Video Service の使用開始](#) - 「ステップ 2: IAM アクセス許可の設定」にある IAM ポリシーが更新され、3 つの IVS アクセス許可 (GetStream、GetStreamSession、ListStreamSessions) と cloudwatch:GetMetricData が追加されました。
- Amazon による Amazon IVS のモニタリング CloudWatch — 4 つの高解像度メトリクス (IngestAudioBitrate、IngestFramerateIngestVideoBitrate、) が追加されましたKeyframeInterval。

2023 年 10 月 12 日の更新: この CloudWatch ドキュメントは削除され、コンテンツは [「IVS 低レイテンシーストリーミングのモニタリング」](#) に移動されました。

- [Amazon IVS EventBridge での Amazon の使用](#) - セッション作成とセッション終了の 2 つのイベントを追加しました。
- [Amazon IVS API リファレンス](#) — 多くの点が変更されました。
 - 2 つの新しいエンドポイント: GetStreamSession および ListStreamSessions。
 - 7 つの新しいオブジェクト: AudioConfiguration IngestConfiguration、StreamEvent StreamFilters、StreamSession、StreamSessionSummary、および VideoConfiguration。
 - ストリームおよび StreamSummary オブジェクトの新しいフィールド (streamID)。これは、GetStream および ListStreams レスポンスにも影響します。
 - リクエストの新しいフィールド (filtersBy) ListStreams。

2021 年 10 月 20 日

Amazon IVS Broadcast SDK: Android 1.1.0 および iOS

プラットフォーム	ダウンロードおよび変更
すべて	<ul style="list-style-type: none"> • トランジションメソッドに提供されるスロット設定の名前が target-slot name パラメータと一致しない場合に、ミキサースロット構成が予期しない状態になるバグを修正しました。 • まれなクラッシュの発生を減らし、安定性を向上させました。 • プリセットのビットレートを再調整して、期待されるユーザーエクスペリエンスをよりよく反映します。これらはブロードキャスト SDK のリファレンスドキュメントに記載されています。 • スタンダード (ポートレート/ランドスケープ) — 初期: 2.1 Mbps。最大: 6 Mbps。 • ベーシック (ポートレート/ランドスケープ) — 初期: 1.2 Mbps。最大: 1.5 Mbps。 • ゲーム (ポートレート/ランドスケープ) (Android のみ) — 初期: 2.1 Mbps。最大 6 Mbps。 • モノラルオーディオのサポートが追加されました。ブロードキャストセッションは、1 つまたは 2 つのオーディオチャネル (それぞれモノラルまたはステレオ) で構成できるようになりました。また、カスタムオーディオソースは、1 つまたは 2 つのオーディオチャネルで構成できます。 • ミキサーのキャンバスとスロットの起点を左上に変更しました。これは、開発者にとってより自然で、より一貫したユーザビリティを

プラットフォーム	ダウンロードおよび変更
	<p>提供する必要があります。カスタムミキサー スロットを使用している場合は、コードを更新する必要があります。 Broadcast SDK ミキサー:1.0.0 から 1.1.0 への移行を参照してください。</p> <ul style="list-style-type: none">• 新しいドキュメントページ「Amazon IVS へのブロードキャスト: ミキサーガイド」を追加しました。
Android Broadcast SDK 1.1.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.1.0/android/</p> <ul style="list-style-type: none">• デバイスの方向の変更によって SDK がクラッシュする可能性があるバグを修正しました。• <code>getPreviewView()</code> が最初に呼び出されたときにのみ機能するバグを修正しました。これにより、<code>getPreviewView()</code> は呼び出されるたびに新しい <code>ImagePreviewView</code> を返すため、同じデバイスまたはセッションの複数の <code>ImagePreviewViews</code> を同時にビュー階層に追加できるようになりました。多くの <code>ImagePreviewViews</code> を同時に使用すると、パフォーマンスが低下する可能性があることに注意してください。• ブロードキャストセッション全体を解放せずにシステムキャプチャサービスを停止するために <code>stopSystemCapture()</code> を追加しました。• デバイスを接続するときにミキサー スロットの優先デバイスを無視するために、<code>attachDevice</code> オーバーライドを追加しました。

プラットフォーム	ダウンロードおよび変更
iOS Broadcast SDK 1.1.0	<p>ダウンロード : https://broadcast.live-video.net/1.1.0/AmazonIVSBroadcast.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.1.0/ios/</p> <ul style="list-style-type: none">• <code>IVSMixerSlotConfiguration</code> オブジェクトに <code>size</code> を設定すると、<code>matchCanvasSize</code> が <code>false</code> に設定されるようになりました。同様に、<code>IVSMixerSlotConfiguration</code> オブジェクトに <code>aspect</code> を設定すると、<code>matchCanvasAspectMode</code> が <code>false</code> に設定されます。• 事前エンコードされた動画を使用したバックグラウンドオーディオのサポートが追加されました。新しいメソッド <code>createAppBackgroundImageSourceOnComplete</code> では、アプリをバックグラウンド処理するときのデフォルトの動作を変更します。以前は、SDK がカメラまたは GPU にアクセスできなくなったため、ストリーム全体が停止しました (つまり、動画入力の合成や動画エンコーディングは実行できませんでした)。 <p>新しいメソッドは、<code>IVSCustomVideoSource</code> のサブクラスを返します。通常、<code>IVSCustomVideoSource</code> を使用すると、ブロードキャストする画像サンプルを送信できません。サブクラスを使用すると、アプリがバックグラウンドにあるときに、後でブロードキャストするために事前にエンコードされる画像サンプルを送信できます。</p>

Broadcast SDK ミキサー:1.0.0 から 1.1.0 への移行

Broadcast SDK のバージョン 1.1.0 では、ミキサーの座標系の動作が変更されます。1.0.0 では、ミキサーが矛盾する原点を使用しました。1.1.0 では、原点は左上隅です。新しいものを見る [Amazon IVS Broadcast SDK ミキサーガイド](#)。

キャンバスの変更:水平 (X 軸) の位置は変更されません。1.0.0 と比較して、垂直方向の配置が反転されます。Y 軸の値を 0 に設定すると、スロットはキャンバスの上部 (1.0.0 の場合と同様に、下ではなく) に配置されます。スロットを 1.0.0 と同じ位置に保つには、キャンバスの高さから現在の Y 値を減算します。 $\text{config.video.size.height} - y$

スロットの変更:1.1.0 ではスロットの左上の原点もあります。方向は 1.0.0 から変更されませんが、原点は中心から左上にシフトしています。左上に配置されたスロットは (0, 0) になり、右下に整列したスロットは次のようになります。

$(\text{canvas_width} - \text{slot_width}, \text{canvas_height} - \text{slot_height})$

スロットを 1.0.0 と同じ位置に保つには、X 位置から幅の半分、Y 位置から高さの半분을減算します。また、スロットのサイズは左上隅を基準にしています。したがって、スロットを中央から拡張するには、サイズと同時に位置を変更する必要があります。そうしないと、スロットが下がり、右に成長しているように見えます。

Broadcast SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	2.23 MB	5.75 MB
armeabi-v7a	2.07 MB	4.99 MB
x86_64	2.35 MB	5.78 MB
x86	2.55 MB	6.78 MB

Broadcast SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64	3.11 MB	6.74 MB

2021 年 9 月 29 日

Amazon IVS Player SDK: Android 1.5.1

プラットフォーム	ダウンロードおよび変更
Android プレイヤー 1.5.1	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.5.1/android/</p> <ul style="list-style-type: none"> FixedgetVersion() となり、正しいバージョン番号が返されます。

Mobile SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.00 MB	2.80 MB
armeabi-v7a	0.83 MB	2.15 MB
x86_64	1.11 MB	3.07 MB
x86	1.12 MB	2.94 MB

2021 年 9 月 28 日

Amazon IVS Player SDK 1.5.0

プラットフォーム	ダウンロードおよび変更
すべて	<ul style="list-style-type: none"> ネットワークの再接続中にプレイメソッドが呼び出されずに、ロードされたストリームが play になる可能性がある問題を修正しました。 ストリームが切断された後、プレーヤーが PLAYING 状態に移行するのではな

プラットフォーム	ダウンロードおよび変更
	<p>く、ENDED 状態のままになる問題を修正しました。</p> <ul style="list-style-type: none">• より多くのエンコーダをサポートするために、CEA-608 キャプション解析を更新しました。• パススルーコンテンツを再生するプレーヤーの能力を向上させました。つまり、BASIC チャンネルからのコンテンツと STANDARD チャンネルからの最高品質です。

プラットフォーム	ダウンロードおよび変更
<p>Web プレイヤー 1.5.0 および Video.js 統合 および JW Player 統合</p>	<p>NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player</p> <p>スクリプトアセット: https://player.live-video.net/1.5.0/amazon-ivs-player.min.js</p> <p>Video.js テクニカルアセット: https://player.live-video.net/1.5.0/amazon-ivs-videojs-tech.min.js</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.5.0/web/</p> <ul style="list-style-type: none">• プレイヤーが動画デコードおよびプレイリストネットワークエラーから回復する方法を改善しました。• ネイティブHTML5コントロールが有効になっているときに、ライブストリームが再開しない (または遅延後に再開される) バグを修正しました。• <code>getBuffered ()</code> メソッドが期待値ではなく未定義を返す問題を修正しました { <code>start: 0, end: 0</code> } コンテンツが読み込まれない場合。• Video.js の picture-in-picture モードのサポートを追加しました。• プレイヤーのデフォルトのログレベルを「警告」ではなく「エラー」に変更しました。

プラットフォーム	ダウンロードおよび変更
Android プレイヤー 1.5.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.5.0/android/</p> <ul style="list-style-type: none"> • Android SDK 30 エミュレータで発生するズームインのバグを修正しました。 • のパフォーマンスが改善されました。PlayerView レイアウトを表示します。 • 1.5.0 の代わりに、<code>getVersion()</code> が <code>1.5.0-ivs.rc.2</code> を返します。
iOS プレイヤー 1.5.0	<p>ダウンロード : https://player.live-video.net/1.5.0/AmazonIVSPlayer.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.5.0/ios/</p> <ul style="list-style-type: none"> • Apple Silicon MacでiOSシミュレータのサポートを追加しました。 • プレーヤーの割り当てを解除するまで再生中のプレーヤーのメモリヒープが増加し続ける問題を修正しました。 • 動画に不良データがあるときの再生動作を改善。無視して再生を停止するのではなく、再生を続行します。

Mobile SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.00 MB	2.80 MB
armeabi-v7a	0.83 MB	2.15 MB

アーキテクチャ	圧縮サイズ	非圧縮サイズ
x86_64	1.11 MB	3.07 MB
x86	1.12 MB	2.94 MB

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
universal	0.92 MB	1.99 MB
arm64	0.47 MB	1.09 MB
armv7	0.46 MB	0.87 MB

2021 年 9 月 8 日

Amazon IVS Player SDK 1.4.1

プラットフォーム	ダウンロードおよび変更
すべて	順番通りに挿入されなかったキャプションを処理するようにクローズドキャプションデコーダを修正しました。
Web プレイヤー 1.4.1 および Video.js 統合 および JW Player 統合	<p>NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player</p> <p>スクリプトアセット: https://player.live-video.net/1.4.1/amazon-ivs-player.min.js</p> <p>Video.js テクニカルアセット: https://player.live-video.net/1.4.1/amazon-ivs-videojs-tech.min.js</p>

プラットフォーム	ダウンロードおよび変更
	リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.4.1/web/
Android プレイヤー 1.4.1	リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.4.1/android/
iOS プレイヤー 1.4.1	ダウンロード : https://player.live-video.net/1.4.1/AmazonIVSPlayer.xcframework.zip リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.4.1/ios/

Mobile SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.00 MB	2.79 MB
armeabi-v7a	0.83 MB	2.15 MB
x86_64	1.11 MB	3.06 MB
x86	1.11 MB	2.94 MB

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
universal	0.89 MB	1.91 MB
arm64	0.45 MB	1.05 MB
armv7	0.44 MB	0.84 MB

2021 年 8 月 13 日

ListTagsForResource API エンドポイント

このエンドポイントのページ割り (maxResults リクエストフィールドと nextToken リクエスト/レスポンスフィールド) のサポートを削除しました。(ページ割りは正常に機能していませんでした。)

2021 年 8 月 10 日

Amazon IVS Player SDK 1.4.0

プラットフォーム	ダウンロードおよび変更
すべて	<ul style="list-style-type: none">• DURATION_CHANGED イベントまたは READY 状態の更新の直後にシークが発生した場合に、VODの再生が停止する稀に発生する問題を修正しました。• ID3 メタデータでストリーム再生時のメモリリークを修正しました。• 挿入したキャプションが正しくレンダリングされない場合のエッジケースを修正しました。• プレーヤーのアダプティブビットレートストリーミングアルゴリズムのパフォーマンスを改善しました。• まれなクラッシュの発生を減らし、プレーヤーの安定性を向上させました。• プレーヤーが作成したスレッドとは異なるスレッドからアクセスした場合のログ警告メッセージを追加しました。• サーバーからプレーヤーへのレイテンシーの計算方法をより詳しく説明するために、getLiveLatency() のドキュメントを更新しました。

プラットフォーム	ダウンロードおよび変更
<p>Web プレーヤー 1.4.0 および Video.js 統合 および JW Player 統合</p>	<p>NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player</p> <p>スクリプトアセット: https://player.live-video.net/1.4.0/amazon-ivs-player.min.js</p> <p>Video.js テクニカルアセット: https://player.live-video.net/1.4.0/amazon-ivs-videojs-tech.min.js</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.4.0/web/</p> <ul style="list-style-type: none">• iOS Safari で TIMED_METADATA イベントが発生しなかったエッジケースを修正しました。• Firefox で低レイテンシーのストリームを再生するときのプレーヤーのアダプティブビットレートストリーミングアルゴリズムのパフォーマンスが向上しました。• ライブストリームに対して常に Infinity を返す <code>getDuration()</code> のドキュメントを修正しました。• デスクトップ Safari で自動再生に失敗する場合があります。不具合を修正しました。• デベロッパーコンソールでの「Cannot read property 'collectLogs' of undefined (未定義のプロパティ 'CollectLogs' を読み取ることができない)」というエラーを修正しました。• Video.js: モードのサポートを追加しました picture-in-picture。• Web: 新しいメソッド <code>setRequestCredentials</code> を追加しました。これは、プレーヤーがクロスオリジンエンドポイントに認証情報リクエストを行うかどうか

プラットフォーム	ダウンロードおよび変更
	<p>かを制御します。リモートエンドポイントは、適切な CORS 応答ヘッダー (リクエストの Origin に一致する <code>Access-Control-Allow-Origin</code> など) で応答する必要があります。 <code>Access-Control-Allow-Credentials</code> は <code>true</code> でなければなりません。</p> <p>この設定は、プレーヤーインスタンスのライフサイクル全体を通して維持されます。したがって、URL エンドポイントを使用する後続のすべての <code>player.load()</code> の呼び出しは、適切な CORS ヘッダーで応答する必要があります。</p> <p>このメソッドは、iOS ブラウザプラットフォームには影響しません。iOS プラットフォームで認証されたクロスオリジンリクエストを許可するには、ユーザーが明示的にクロスサイトトラッキングを許可し、Cookie を許可する必要があります。これらは、デバイスおよび各ブラウザアプリに設定されます。</p>

プラットフォーム	ダウンロードおよび変更
Android プレイヤー 1.4.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.4.0/android/</p> <ul style="list-style-type: none">• 高解像度のポートレート動画がデバイスでサポートされていても、サポートされていない動画と見なされていた問題を修正しました。• 特定の Android デバイスで再生レートを変更できない問題を修正しました。• 出力サーフェスが設定されていない場合にコンテンツをデコードしないように、バックグラウンドの動画処理を更新しました。• <code>player.release()</code> メソッドが呼び出された後に SDK 呼び出しを無視する追加のチェック機能を実装しました。この機能により、プレイヤーの安定性が向上します。• 最適化により Android ライブラリのファイルサイズが小さくなりました。

プラットフォーム	ダウンロードおよび変更
iOS プレイヤー 1.4.0	<p>ダウンロード : https://player.live-video.net/1.4.0/AmazonIVSPlayer.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.4.0/ios/</p> <ul style="list-style-type: none">• プレーヤーの割り当てを解除するまで再生中のプレーヤーのメモリヒープが増加し続ける問題を修正しました。• 動画再生の上にアニメーション (UI アニメーションや GIF 画像など) がある場合に発生する可能性のあるデッドロックを修正しました。• メディアサービスのリセットイベント中に発生する可能性のあるクラッシュを修正しました。• 品質を切り替え中に発生する可能性があった <code>CMFormatDescriptionRef</code> のメモリリークを解決しました。• <code>IVSPlayerView</code> クラスや <code>IVSPlayerLayer</code> クラスの IVS 固有のプロパティがメインスレッド以外のスレッドでアクセスされた場合にログ記録するエラーメッセージを追加しました。• 出力サーフェスが設定されていない場合にコンテンツをデコードしないように、バックグラウンドの動画処理を更新しました。• IOS SDK リファレンスのドキュメントカバーレッジを改善しました。• 最適化により iOS ライブラリファイルサイズが小さくなりました。

Mobile SDK サイズ: Android

アーキテクチャ	圧縮サイズ	非圧縮サイズ
arm64-v8a	1.00 MB	2.79 MB
armeabi-v7a	0.83 MB	2.15 MB
x86_64	1.11 MB	3.06 MB
x86	1.11 MB	2.93 MB

Mobile SDK サイズ: iOS

アーキテクチャ	圧縮サイズ	非圧縮サイズ
universal	0.89 MB	1.91 MB
arm64	0.45 MB	1.05 MB
armv7	0.44 MB	0.84 MB

2021 年 7 月 27 日

Amazon IVS Broadcast SDK: Android 1.0.0 および iOS 1.0.0

プラットフォーム	ダウンロードおよび変更
Android Broadcast SDK 1.0.0	リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.0.0/android/
iOS Broadcast SDK 1.0.0	ダウンロード: https://broadcast.live-video.net/1.0.0/AmazonIVSBroadcast.xcframework.zip リファレンスドキュメント: https://aws.github.io/amazon-ivs-broadcast-docs/1.0.0/ios/

2021 年 6 月 1 日

Amazon IVS Player SDK: Android 1.3.3 と iOS 1.3.3

プラットフォーム	ダウンロードおよび変更
Android および iOS	高解像度のポートレート動画がサポートされていても、デバイスがサポートされていなかったとみなされていた問題を修正しました。
Android プレイヤー 1.3.3	リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.3.3/android/
iOS プレイヤー 1.3.3	ダウンロード : https://player.live-video.net/1.3.3/AmazonIVSPlayer.xcframework.zip リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.3.3/ios/ <ul style="list-style-type: none">• 動画再生の上にアニメーション (UI アニメーションや GIF 画像など) がある場合に発生する可能性のあるデッドロックを修正しました。• メディアサービスのリセットイベント中に発生する可能性のあるクラッシュを修正しました。

2021 年 5 月 19 日

Amazon IVS Player SDK: Android 1.3.2

リファレンスドキュメント: <https://aws.github.io/amazon-ivs-player-docs/1.3.2/android/>

プレイヤーの安定性を向上させるために、`player.release()` メソッドが呼び出された後に API 呼び出しを無視する追加のチェック機能を実装しました。

2021 年 5 月 5 日

Amazon IVS Player SDK 1.3

プラットフォーム	ダウンロードおよび変更
すべて	<ul style="list-style-type: none">• TextCue 使用状況ドキュメントの使用に関する SDK ドキュメントを更新しました。プレイヤー SDK に関する最新のリファレンスについては、Amazon IVS ドキュメントのランディングページを参照してください。• 不正なモノラル入カストリームのオーディオ再生に関する問題を修正しました。• ライブ HLS ウィンドウ以外でコンテンツを再生するときに、まれに発生する可能性がある再生エラーを修正しました。• スタンダードの HLS ライブストリームと録画ストリームを再生するプレイヤー機能が改善されました。• 特に、新規ストリームをロードするときにゼロにリセットされるよう <code>getLiveLatency</code> の精度を改善しました。• 適応ビットレートストリーミング (ABR、adaptive bitrate streaming) アルゴリズムを改善し、ネットワーク接続の改善時に動画品質をより高速に向上させました。• まれなクラッシュの発生を減らし、プレイヤーの安定性を向上させました。
Web プレーヤー 1.3.1 および Video.js 統合 および JW Player 統合	NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player スクリプトアセット: https://player.live-video.net/1.3.1/amazon-ivs-player.min.js

プラットフォーム	ダウンロードおよび変更
	<p>Video.js テクニカルアセット: https://player.live-video.net/1.3.1/amazon-ivs-videojs-tech.min.js</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.3.1/web/</p> <ul style="list-style-type: none">• ロード直後に実行されたシークコールが無視され、プレイヤーが正しくない位置から開始する問題を修正しました。• 録画されたコンテンツ (VOD と呼ばれます) 内でのシークに関する複数の問題を修正しました。• 最適でないネットワーク条件で再生が失敗する可能性がある問題を修正しました。• iOS モバイル Web ブラウザーでの IVS 時間指定メタデータのサポートが追加されました。• デスクトップ Safari で自動再生に失敗する可能性がある不具合を修正しました。• Web SDK getVersion 関数は、プレイヤーのバージョンにハッシュを追加なくなりました。• バッファリングされた範囲の正確な開始位置までシークすると、別のシークフォワードになる可能性がある問題を修正しました。• macOS Safari 14 以降で、低レイテンシーの適応ビットレートストリーミング (ABR、adaptive bitrate streaming) を有効にしました。• 安全でないインポートの副作用を排除して、サーバーコンテキストでプレイヤーをロードする問題を修正しました。

プラットフォーム	ダウンロードおよび変更
	<ul style="list-style-type: none">amazon-ivs-player NPM パッケージを変更し、で使用される LogLevel 列挙型をエクスポートしましたsetLogLevel 。 <p>注: Web プレーヤー 1.3.0 NPM パッケージは存在しますが、動作しません。NPM では非推奨とマークされています。Web プレーヤー 1.3.1 以降を使用してください。</p>

プラットフォーム	ダウンロードおよび変更
Android プレイヤー 1.3.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.3.0/android/</p> <ul style="list-style-type: none">• アプリが Android 11 (API レベル 30) をターゲットとし、ユーザーがモバイルネットワークで Android 11 を実行していた場合、プレイヤー SDK がクラッシュする問題を修正しました。• ネットワーク回復の問題を修正しました。ネットワーク接続が切断されると再生が自動的に一時停止し、接続が復元されると再生が再開するようになりました。Player.Listener の onNetworkUnavailable コールバックを使用して、ネットワーク状態の変化を観察します。• VOD の再生中にプレイヤーのコントロールが <code>setControlsEnabled(false)</code> で非表示にならない問題を修正しました。• クライアントアプリが古い (4.0 より前) バージョンのを使用している場合に SDK がクラッシュすることがある問題を修正しました OkHttp。• Amazon IVS Android プレイヤーライブラリが JCenter リポジトリから Maven Central に移動しました。• ライブラリから BuildConfig バージョンプロパティを削除しました。

プラットフォーム	ダウンロードおよび変更
iOS プレイヤー 1.3.0	<p>ダウンロード: https://player.live-video.net/1.3.0/AmazonIVSPlayer.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.3.0/ios/</p> <ul style="list-style-type: none">• 単一の HLS メディアセグメント内でオーディオサンプルレートの変更があった場合、iOS SDK で正しく処理できない問題を修正しました。これにより、メディア不良によりメモリが予期せず増加し、再生に失敗したり、クラッシュしたりする可能性があります。• ネットワーク回復の問題を修正しました。ネットワーク接続が切断されると再生が自動的に一時停止し、接続が復元されると再生が再開するようになりました。playerNetworkDidBecomeUnavailable デリゲートメソッドを使用して、ネットワーク状態の変化を観察します。• 時間の経過とともにiOSメモリが増加する問題を修正しました。• オーディオハードウェアの問題の正常な処理が追加されました。メディアサービスのリセット通知が発生すると、再生が自動的に一時停止されるようになりました (AVAudioSessionMediaServicesWereReset Notification)。リセット時にメディアが再生されていると、依然として再生エラーが発生する可能性があることに注意してください。• オーディオセッションの中断処理が追加されました。オーディオセッションの中断が始まると、再生が自動的に一時停止するようにな

プラットフォーム	ダウンロードおよび変更
	<p>りました。中断が終了すると、プレイヤーが再生していた場合は自動的に再生が再開され、中断オプションはアプリが再生を再開する必要があることを示します。</p>

2021 年 4 月 26 日

Service Quotas と CloudWatch 使用状況メトリクスの統合

CloudWatch を使用して、使用状況メトリクス を介して CloudWatch サービスクォータをプロアクティブに管理できます。[Amazon IVS Service Quotas](#) を参照してください。

2021 年 4 月 13 日

新しい CloudWatch メトリクス

CloudWatch 同時視聴と同時ストリームに メトリクスが追加されました。「Amazon による Amazon IVS のモニタリング」を参照してください CloudWatch。

2023 年 10 月 12 日の更新: この CloudWatch ドキュメントは削除され、コンテンツは [「IVS 低レイテンシーストリーミングのモニタリング」](#) に移動されました。

関連する Service Quotas の名前が、新しいメトリクスに一致するよう更新されました。「[サービスクォータ \(低レイテンシーストリーミング\)](#)」を参照してください。

「視聴」の完全な定義については、[Amazon IVS Glossary](#) を参照してください。

2021 年 4 月 7 日

S3 への自動録画 (R2S3)

Amazon IVS では、ライブ動画コンテンツを Amazon S3 に保存できるようになりました。保存した動画は、後で編集や VOD としての再生などの操作に使用できます。

チャンネルの録画を有効にすると、チャンネルのすべてのライブブロードキャストは自分の使いたい S3 バケットに保存できます。使用可能なすべての画質のレンディションとサムネイルイメージが保存されます。録画設定も保存されるため、追加チャンネルに簡単に再利用できます。

Amazon IVS コンソールまたは API を使用して録画設定をセットアップし、録画を有効/無効にできます。詳細については、[IVS の開始方法](#)と [Amazon IVS API リファレンス](#)を参照してください。

2021 年 1 月 28 日

Amazon IVS Player SDK: JW Player 統合 1.2.0

Amazon IVS プレイヤーと JW Player が統合されました。「[JW Player 統合](#)」を参照してください。

既知の問題: 動画の再生時間が 00:00 と表示され、シークバーの再生ヘッドをドラッグしてもシークしない場合があります。これは、iPhone で Safari を使用して Amazon IVS ライブストリームと VOD が混在した広告なしのプレイリストを視聴している場合にのみ発生します。

2020 年 12 月 16 日

Amazon IVS Player: SDK for Android 1.2.1

リファレンスドキュメント: <https://aws.github.io/amazon-ivs-player-docs/1.2.1/android/>

このリリースには、問題を修正する Android プレイヤーのパッチが含まれています。以前の Android プレイヤーの SDK リリースでは、アプリが Android 11 (API レベル30) を対象とし、ユーザーがモバイルネットワークで Android 11 を実行している場合、SDK がクラッシュします。

2020 年 11 月 23 日

Amazon IVS Player SDK 1.2.0

プラットフォーム	ダウンロードおよび変更
すべて	Amazon IVS ストリームの検出が改善され、メトリクスがより正確になりました。
Web プレイヤー 1.2.0 および Video.js 統合	NPM パッケージ: https://www.npmjs.com/package/amazon-ivs-player スクリプトアセット: https://player.live-video.net/1.2.0/amazon-ivs-player.min.js

プラットフォーム	ダウンロードおよび変更
	<p>Video.js テクニカルアセット: https://player.live-video.net/1.2.0/amazon-ivs-videojs-tech.min.js</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.2.0/web/</p> <ul style="list-style-type: none">• ストリームのマスタープレイリストが使用できない場合は、すべてのウェブ再生ソース <code>ErrorNotAvailable</code> に対して を発行します。• 同時視聴者 (CCV) の制限到達に関連するエラーについてのリファレンスドキュメントを更新しました。
Android プレイヤー 1.2.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.2.0/android/</p> <ul style="list-style-type: none">• Android で <code>getSessionId</code> 関数がクラッシュする問題を修正しました。• 同時視聴者 (CCV) の制限到達に関連するエラーについてのリファレンスドキュメントを更新しました。 <p>既知の問題: アプリが Android 11 (API レベル 30) をターゲットとし、ユーザーがモバイルネットワークで Android 11 を実行している場合、プレイヤー SDK がクラッシュします。この問題は、次のリリースで修正されます。それまでは、以前の Android API レベル (29 以下) をターゲットにすることをお勧めします。</p>

プラットフォーム	ダウンロードおよび変更
iOS プレイヤー 1.2.0	<p>ダウンロード: https://player.live-video.net/1.2.0/AmazonIVSPlayer.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.2.0/ios/</p> <ul style="list-style-type: none">• ストリーム URL を切り替えたり、プレイヤーを閉じる際に、メモリが破損する可能性のある原因を修正します。• 再生を開始する前に最適なオーディオピッチ補正を有効にできなかった場合に、再生が失敗する問題を解決します。ピッチ補正は、通常よりも高速または低速の再生速度で音質を向上させます。ピッチ補正を有効にできない場合、または最高品質の補正アルゴリズムが使用できない場合、メッセージが記録されますが、再生は続行されます。 <p>既知の問題: 単一の HLS メディアセグメント内でオーディオサンプルレートの変更がある場合、iOS SDK はそれを正しく処理できません。これにより、メモリが予期せず増加し、再生に失敗したり、メディア不良が原因でクラッシュしたりする可能性があります。この問題は、iOS プレイヤー の次のメジャーリリースで修正されます。</p>

2020 年 11 月 12 日

新しいイベントフィールド、stream_id

stream_id フィールドが複数のイベントに追加されました。「[IVS EventBridge での Amazon の使用](#)」を参照してください。

2020 年 11 月 9 日

コンソールへのメタデータ表示の追加

時間指定メタデータを Amazon IVS コンソールから表示できるようになりました。Amazon IVS ユーザーガイドの [Embedding Metadata within a Video Stream](#) に記載された [時間指定メタデータの表示](#) の新しいセクションを参照してください。

2020 年 10 月 30 日

CloudFormation サポート

Amazon IVS が AWS をサポートするようになりました CloudFormation。これにより、Amazon IVS のお客様は、AWS でチャンネル、ストリームキー、再生キーペアを作成および管理できます CloudFormation。

の Amazon IVS サポート CloudFormation は、Amazon IVS が利用可能なすべての [AWS リージョン](#) で利用できます。開始するには、「AWS CloudFormation ユーザーガイド」の [「Amazon IVS 製品ページ」](#) または [「Amazon IVS 情報」](#) を参照してください。

2020 年 10 月 27 日

チャンネル、CCV、CCB 制限の引き上げ

3 つのサービスクォータが引き上げられました。

- ユーザーが作成できるチャンネルの最大数が、AWS リージョンごとに、500 から 5,000 に増加しました。
- AWS リージョンの全チャンネルでライブチャンネルの再生が可能な同時視聴者の最大数が、3,000 から 15,000 に増加しました。
- 同時ブロードキャストの最大数 (同時にストリーミングできるチャンネル) が、AWS リージョンごとに 30 から 100 に増加しました。

これらの増加は、Amazon IVS が使用可能な [すべてのリージョン](#) で利用できます。詳細については、Amazon IVS ユーザーガイドの [サービスクォータ \(低レイテンシーストリーミング\)](#) を参照してください。

2020 年 10 月 9 日

新しい Service Quotas と EventBridge イベント

取り込み解像度に関連するサービスクォータと EventBridge イベントが追加されました。「[サービスクォータ \(低レイテンシーストリーミング\)](#)」および「[IVS EventBridge での Amazon の使用](#)」を参照してください。

Amazon IVS Player: SDK for Web 1.1.2

NPM パッケージ : <https://www.npmjs.com/package/amazon-ivs-player>

スクリプトアセット: <https://player.live-video.net/1.1.2/amazon-ivs-player.min.js>

Video.js テクニカルアセット: <https://player.live-video.net/1.1.2/amazon-ivs-videojs-tech.min.js>

リファレンスドキュメント: <https://aws.github.io/amazon-ivs-player-docs/1.1.2/web/>

このリリースには、Microsoft Edge を使用している視聴者に影響を与える問題を修正する Web Player パッチが含まれています。ストリームの自動品質モードがオンになっている (ABR が有効になっている) 場合、Microsoft Edge を使用している視聴者の低レイテンシー再生が機能せず、ストリームの再生におけるレイテンシーが高くなっていました。

2020 年 10 月 7 日

Amazon IVS Player SDK 1.1.0

Amazon Interactive Video Service (IVS) プレイヤー SDK では[セマンティックバージョニング](#)を使用します。

プラットフォーム	ダウンロードおよび変更
すべて	<ul style="list-style-type: none">プレイヤーの適応ビットレートアルゴリズムで、品質が誤って 160p に低下する場合があります。問題を修正しました。再生可能な動画品質でない場合、プレイヤーがエラーをスローします。

プラットフォーム	ダウンロードおよび変更
	<ul style="list-style-type: none">• VOD シーク動作を更新: 最後までシークしようとする、プレイヤーはエラーを返す代わりに最後までシークします。• プレイヤーは、エラー回復中に利用可能なすべての品質を使い果たした後、致命的なエラーをスローします。

プラットフォーム	ダウンロードおよび変更
Web プレイヤー 1.1.0	<p>NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player</p> <p>スクリプトアセット: https://player.live-video.net/1.1.0/amazon-ivs-player.min.js</p> <p>Video.js テクニカルアセット: https://player.live-video.net/1.1.0/amazon-ivs-videosjs-tech.min.js</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.1.0/web/</p> <p>既知の問題:</p> <ul style="list-style-type: none">• Video.js が利用できない場合、<code>registerIVSQualityPlugin</code> は <code>console.error</code> への書き込みの代わりに例外をスローするようになりました。• <code>registerIVSTech</code> または <code>registerIVSQualityPlugin</code> が複数回呼び出された場合、最初の呼び出し後の呼び出しに対して (再登録を試行する代わりに) 何も行いません。• 最初のパラメータから <code>registerIVSQualityPlugin</code> への型が <code>VideoJS</code> から <code>any</code> に変更されました。• サーバー側のレンダリングを有効にするために、ブラウザコンテキストの依存関係を削除しました。• ミュート解除に応答してブラウザが自動一時停止した場合、プレイヤーは <code>AUDIO_BLOCKED</code> イベントを起動し、ミュートされた再生を再開します。

プラットフォーム	ダウンロードおよび変更
	<ul style="list-style-type: none"> • ネットワーク接続回復が追加されました。ネットワークタイムアウトは、クライアントアプリにエラー状態を送信しません。代わりに、ネットワーク接続が失われた場合: <ul style="list-style-type: none"> • アプリが再生中の場合、プレイヤーライブラリは NETWORK_UNAVAILABLE イベントをアプリに送信し、プレイヤーは IDLE 状態になります。接続が回復すると、プレイヤーライブラリの再生が再開し、アプリは PLAYING イベントを受け取ります。 • アプリが一時停止している場合、NETWORK_UNAVAILABLE イベントはアプリに送信されず、プレイヤーライブラリは IDLE 状態のままになります。接続が回復すると、プレイヤーライブラリは IDLE 状態のままになります。 • アプリが再生しようとする場合は常に、プレイヤーライブラリは通常の再生を試行します。NETWORK_UNAVAILABLE イベントがアプリに送信され、プレイヤーは IDLE 状態になります。
Android プレイヤー 1.1.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.1.0/android/</p> <p>既知の問題: アプリが Android 11 (API レベル 30) をターゲットとし、ユーザーがモバイルネットワークで Android 11 を実行している場合、プレイヤー SDK がクラッシュします。この問題は、次のリリースで修正されます。それまでは、以前の Android API レベル (29 以下) をターゲットにすることをお勧めします。</p>

プラットフォーム	ダウンロードおよび変更
iOS プレイヤー 1.1.0	<p>ダウンロード: https://player.live-video.net/1.1.0/AmazonIVSPlayer.xcframework.zip</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.1.0/ios/</p> <ul style="list-style-type: none"> UIKit からの次のメッセージが表示され、クラッシュを引き起こす可能性のある問題を修正しました。「Modifications to the layout engine must not be performed from a background thread after it has been accessed from the main thread. (メインスレッドからアクセスした後、レイアウトエンジンへの変更をバックグラウンドスレッドから実行しないでください)」実行すると、アプリケーションのバックグラウンド処理およびフォアグラウンディング処理時にクラッシュが発生する可能性があります。

2020 年 9 月 14 日

新しいイベントフィールド、channel_name

channel_name フィールドが複数のイベントに追加されました。「[IVS EventBridge での Amazon の使用](#)」を参照してください。

2020 年 8 月 19 日

再生承認 (プライベートチャンネル)

Amazon IVS から、プライベートチャンネルを作成する機能が提供され、ストリームを視聴する視聴者を制限できるようになりました。チャンネルの再生承認を有効にして、承認された再生リクエスト用の署名付き JSON Web Token (JWT) を生成することにより、動画再生へのアクセスを管理します。詳細については、「[プライベートチャンネルの設定](#)」を参照してください。

チャンネルオブジェクトの新しい `authorized` フィールドは、チャンネルがプライベートかどうかを示します。[Amazon IVS API リファレンス](#)を参照してください。

2020 年 8 月 11 日

Amazon IVS Player SDK: SDK for iOS 1.0.6

ダウンロード: <非推奨>

リファレンスドキュメント: <https://aws.github.io/amazon-ivs-player-docs/1.0.6/ios/>

このリリースには、iOS プレイヤーパッチが含まれています。このパッチは、一部の iOS プレイヤーアプリが Apple App Store に追加されない問題を修正します。具体的には、ビットコードを有効にして構築されたアプリが、アップロード後に App Store Connect の検証に失敗する問題です。

2020 年 8 月 5 日

Amazon IVS EventBridge での Amazon の使用

Amazon IVS EventBridge イベントが Amazon EventBridge コンソールから利用可能になりました。[Amazon IVS の Amazon EventBridge ルールの作成](#) 「Amazon IVS ユーザーガイド」の「Amazon IVS EventBridge で Amazon を使用する」のセクションを参照してください。

2020 年 7 月 15 日

プレイヤーバージョン 1.0

Amazon Interactive Video Service (IVS) プレイヤー SDK では[セマンティックバージョンニング](#)を使用します。

プラットフォーム	ダウンロードおよび変更
すべて	既知の問題: <code>setAutoMaxQuality</code> や <code>setQuality</code> 関数を使用すると、指定した品質が現在のストリームに正しく適用されませんが、新しいストリームをロードすると正しく適用されません。これを回避するには、新しい

プラットフォーム	ダウンロードおよび変更
Web プレイヤー 1.0.0	<p>ストリームをロードするときに、PlayerState.READY の後に新しいストリーム用の品質で呼び出します。</p> <p>NPM パッケージ : https://www.npmjs.com/package/amazon-ivs-player</p> <p>スクリプトアセット: https://player.live-video.net/1.0.0/amazon-ivs-player.min.js</p> <p>Video.js テクニカルアセット: https://player.live-video.net/1.0.0/amazon-ivs-videojs-tech.min.js</p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.0.0/web/</p> <p>既知の問題:</p> <ul style="list-style-type: none">• iOS モバイルブラウザ (Safari や Chrome など) による VOD 再生時に、後方シークするとプレイヤーがミュートになります。これを回避するには、シーク後に <code>player.setMuted(false)</code> を呼び出します。• iOS モバイルブラウザによる VOD 再生時に、目的の位置を直接選択すると後方シークが断続的に動作します。これを回避するには、シークバーを目的の位置にドラッグします。• Video.js 統合を使用して iOS モバイルブラウザで VOD を再生すると、再生ボタンが正しく動作しません。これを回避するには、Video.js を初期化するときに再生ボタンを非表示にします。 https://docs.videojs.com/tutorial-components.html#play-toggle

プラットフォーム	ダウンロードおよび変更
Android プレイヤー 1.0.0	<p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.0.0/android/</p> <p>既知の問題: バックグラウンディングとフォアグラウンディングは、Android 上で VOD 再生のためのオーディオ/動画の同期解除を引き起こす可能性があります。</p>
iOS プレイヤー 1.0.0	<p>ダウンロード: <非推奨></p> <p>リファレンスドキュメント: https://aws.github.io/amazon-ivs-player-docs/1.0.0/ios/</p> <p>既知の問題:</p> <ul style="list-style-type: none">• バックグラウンディングとフォアグラウンディングは、ライブと VOD の再生エラーの原因となります。これを回避するには、UIApplicationDidEnterBackgroundNotification を受信し、UIApplicationDidBecomeActiveNotification で再生が再開されたときにストリームを一時停止します。• iOS 10 デバイスは、バックグラウンドから戻ったときにクラッシュすることがあります。これを回避するには、バックグラウンドに移行する前にプレイヤーの player プロパティを nil に設定します。

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。